

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники
 Направление подготовки 09.04.01 «Информатика и вычислительная техника»
 Отделение школы (НОЦ) информационных технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка и тестирование системы автоматизированного управления WEB-SCADA компании ООО «Элком+»

УДК 004.777:004.455.1.054

Студент

Группа	ФИО	Подпись	Дата
8BM81	Обложенко Алёна Дмитриевна		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Кочегурова Е. А.	к.т.н., доцент		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Конотопский В. Ю.	к.э.н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД ШБИП	Горбенко М. В.	к.т.н		

ДОПУСТИТЬ К ЗАЩИТЕ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ООП	Кочегурова Е. А.	к.т.н., доцент		

Томск – 2020 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

по направлению 09.04.01 «Информатика и вычислительная техника»

Код результата	Результат обучения (выпускник должен быть готов)
Общепрофессиональные компетенции	
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
Профессиональные компетенции	
P5	Выполнять инновационные инженерные проекты по разработке аппаратных и программных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
P6	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
P7	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации аппаратных и программных средств автоматизированных систем различного назначения.

	Общекультурные компетенции
P8	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
P9	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
P10	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P11	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Информационных технологий и робототехники
Направление подготовки (специальность) 09.04.01 Информатика и вычислительная техника Отделение школы (НОЦ) Информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП
_____ Кочегурова Е. А.
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ на выполнение выпускной квалификационной работы

В форме

магистерской диссертации
(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8BM81	Обложенко Алёне Дмитриевне

Тема работы:

Разработка и тестирование системы автоматизированного управления WEB-SCADA компании ООО «Элком+»	
Утверждена приказом директора (дата, номер)	№ 59-63/с от 28.02.2020 г.

Срок сдачи студентом выполненной работы:	01.06.2020 г.
--	---------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе (наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</p>	<p>Объектом исследования является программное обеспечение, предназначенное для разработки или поддержания работы в реальном времени систем сбора, обработки и отображения информации об объекте управления и мониторинга.</p>
---	---

<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> 1. Обзор предметной области. 2. Обзор и анализ существующих WEB-технологий. 3. Обзор и анализ существующих SCADA-систем. 4. Требования к разрабатываемой системе. 5. Программное обеспечение для автоматизированного управления процессами с применением WEB-технологий. 6. Тестирование системы.
<p>Перечень графического материала <i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> 1. Представление функционирования WEB-технологий. 2. Визуальное представление интерфейсов существующих SCADA-систем. 3. Визуальное представление интерфейса программного обеспечения автоматизированного управления процессами. 4. Визуальное представление автоматизированного тестирования WEB-SCADA
<p align="center">Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i></p>	
<p align="center">Раздел</p>	<p align="center">Консультант</p>
<p align="center">Основная часть</p>	<p align="center">Доцент ОИТ ИШИТР, к.т.н., доцент Кочегурова Е. А.</p>
<p align="center">Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p align="center">Доцент ОСГН ШБИП, к.э.н. Конотопский В. Ю.</p>
<p align="center">Социальная ответственность</p>	<p align="center">Ассистент ООД ШБИП, к.т.н. Горбенко М. В.</p>
<p align="center">Английский язык</p>	<p align="center">Доцент ОИЯ, к.п.н. Сидоренко Т.В.</p>
<p align="center">Названия разделов, которые должны быть написаны на русском и иностранном языках:</p>	
<p align="center">Раздел 2 SCADA-systems</p>	

<p>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</p>	<p align="center">10.03.2020</p>
--	----------------------------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Кочегурова Е. А.	к.т.н., доцент		10.03.2020

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8BM81	Обложенко Алёна Дмитриевна		10.03.2020

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Информационных технологий и робототехники

Направление подготовки (специальность) 09.04.01 Информатика и вычислительная техника

Уровень образования Магистратура

Отделение школы (НОЦ) Информационных технологий

Период выполнения _____ (осенний / весенний семестр 2019/2020 учебного года)

Форма представления работы:

Магистерская диссертация (бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	01.06.2020
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
01.06.20	<i>Основная часть</i>	70
01.06.20	<i>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</i>	10
01.06.20	<i>Социальная ответственность</i>	10
01.06.20	<i>Обязательное приложение на иностранном языке</i>	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Кочегурова Е. А.	к.т.н., доцент		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Кочегурова Е. А.	к.т.н., доцент		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8BM81	Обложенко Алёна Дмитриевна

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Магистратура	Направление/специальность	09.04.01 Информатика и вычислительная техника

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Использовать действующие ценники и договорные цены на потребленные материальные и информационные ресурсы, а также указанную в МУ величину тарифа на эл. энергию
2. Нормы и нормативы расходования ресурсов	—
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Действующие ставки единого социального налога и НДС (см. МУ)

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок	Построение плана-графика выполнения ВКР, составление соответствующей сметы затрат, расчет величины НДС и цены результата ВКР
---	--

Перечень графического материала (с точным указанием обязательных чертежей):

1. График проведения и бюджет НТИ
2. Оценка ресурсной, финансовой и экономической эффективности НТИ

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Конотопский Владимир Юрьевич	К. Э. Н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8BM81	Обложенко Алёна Дмитриевна		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8BM81	Обложенко Алёна Дмитриевна

Школа	Отделение (НОЦ)	ОИТ
Уровень образования	Направление/специальность	09.04.01 Информатика и вычислительная техника
Магистратура		

Тема ВКР:

Разработка и тестирование системы автоматизированного управления WEB-SCADA компании ООО «Элком+»

Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<p>Объектом исследования является программное обеспечение, предназначенное для разработки или поддержания работы в реальном времени систем сбора, обработки и отображения информации об объекте управления и мониторинга.</p> <p>В связи с этим, в работе рассматривается направления и процесс тестирования, проводится анализ разрабатываемого приложения, а также разработка автоматизированного тестирования с помощью инструмента Hermione.</p> <p>Магистерская диссертация выполнялась в десятом корпусе ТПУ (117 аудитория) в отделении информационных технологий.</p>
--	---

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>ГОСТ 12.2.032-78; ГОСТ 12.2.061-81; СанПиН 2.2.2/2.4.1340-03</p>
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	<p>Вредные факторы:</p> <ul style="list-style-type: none"> • нарушение параметров микроклимата; • недостаточное освещение; • повышение шума; • электромагнитные излучения <p>Опасные факторы:</p> <ul style="list-style-type: none"> • опасность поражения электрическим током
3. Экологическая безопасность:	<p>Так как объект исследования данной работы разрабатывался на ПЭВМ, то сама разработка не влечет за собой негативных воздействий на окружающую среду, поэтому создание санитарно-защитной зоны и принятие мер по защите атмосферы, гидросферы, литосферы не являются необходимыми.</p>

4. Безопасность в чрезвычайных ситуациях:	– возможные ЧС: пожар и взрыв на рабочем месте; – наиболее типичная ЧС: пожар
--	--

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Горбенко Михаил Владимирович	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ81	Обложенко Алёна Дмитриевна		

Реферат

Выпускная квалификационная работа содержит 107 страниц, 35 рисунков, 13 таблиц, 44 источников, 1 приложение.

Ключевые слова: автоматизация, программное обеспечение, управление, автоматизированная система управления, тестирование программного обеспечения, промышленность, WEB-приложение, WEB-технологии.

Объектом исследования является программное обеспечение, предназначенное для разработки или поддержания работы в реальном времени систем сбора, обработки и отображения информации об объекте управления и мониторинга.

Предметом исследования является тестирование программного обеспечения для автоматизированного управления технологическими процессами через глобальную сеть Интернет при помощи браузеров.

В магистерской диссертации исследуются особенности разработки и тестирования программного обеспечения для сбора данных и диспетчерского контроля с применением WEB-технологий.

В результате исследования протестирована разрабатываемая SCADA-система с помощью различных видов тестирования. Программное обеспечение обеспечивает управление технологическими и производственными процессами непосредственно через браузеры пользователей.

Область применения: нефтяные, газовые, химические и другие компании промышленной назначения. Объектами применения могут быть распределённые объекты электроэнергетики, кустовые площадки обустройства объектов нефтедобычи и объекты других промышленных отраслей.

Содержание

Введение	13
1 WEB-технологии	15
1.1 Существующие WEB-технологии	15
1.1.1 HTML и CSS	16
1.1.2 JavaScript	17
1.1.3 Фреймворк Quasar	19
1.1.4 Selenium, WebdriverIO и Mocha	20
1.2 Промышленный интернет вещей.....	21
2 SCADA-системы	24
2.1 Существующие SCADA-системы.....	25
2.1.1 SCADA «MasterSCADA» компании ИнСАТ (Россия)	25
2.1.2 SCADA «Trace Mode» компании AdAstrA Research Group (Россия)..	27
2.1.3 SCADA «InTouch» компании Wonderware (США)	29
2.1.4 SCADA «RSView32» компании Rockwell Automation (США).....	31
2.2 Предлагаемое решение.....	33
3 WEB-SCADA компании ООО «Элком+»	35
4 Тестирование WEB-SCADA	44
4.1 Функциональное тестирование.....	47
4.2 Юзабилити-тестирование	50
4.3 Тестирование интерфейса	52
4.4 Нагрузочное тестирование.....	53
4.5 Тестирование безопасности	56
4.6 Автоматизированное тестирование с помощью инструмента Hermione ..	58

5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	64
5.1 Организация и планирование работ	64
5.2 Расчет сметы затрат на выполнение проекта	69
5.3 Оценка экономической эффективности проекта	74
6. Социальная ответственность	76
6.1 Правовые и организационные вопросы обеспечения безопасности	76
6.2 Профессиональная социальная ответственность	78
6.3 Экологическая безопасность	86
6.4 Безопасность в чрезвычайных ситуациях	87
Заключение	91
Список публикаций и выступлений	92
Список использованных источников	93
Приложение А	97

Введение

В современном мире автоматизированные системы управления технологическими процессами в нефтяной, газовой и других отраслях народного хозяйства представляют собой многоуровневые распределенные системы управления, которые реализуют такие основные функции, как управление, регистрацию, регулирование и т. д.

На сегодняшний день процесс анализа такого программного продукта является сложной и актуальной задачей, требующей оперативного подхода для повышения качества диспетчерского контроля. Процесс проверки соответствия заявленных к продукту требований и реально реализованной функциональности осуществляется путем наблюдения за его работой в искусственно созданных ситуациях и на ограниченном определенным образом наборе тестов.

Настоящая магистерская диссертация позволяет решить проблемы контроля качества для промышленных систем автоматизированного управления с помощью WEB-технологий. Помимо этого, работа поможет решению актуальной проблемы импортозамещения зарубежных программных продуктов для сферы промышленности.

Целью данной работы является повышение эффективности и внедрение процесса тестирования с помощью WEB-технологий для программного обеспечения, предназначенного для автоматизированного управления технологическими процессами и называемого WEB-SCADA-системой.

Для этого проводятся исследования современных WEB-технологий и промышленности в целом. Такие решения позволяют расширить задачи интеграции интернет-технологий и всей сферы промышленности, сделать обзор подходов к тестированию программного обеспечения.

Научная новизна заключается в разработке автоматизированного тестирования с помощью инструмента Hermione, который ускоряет автоматизацию процесса запуска тестов и имитирует действия пользователя в браузере.

В результате исследования протестирована SCADA-система, ориентирующаяся на использовании браузеров в качестве клиентского приложения и позволяющая управлять технологическими и производственными процессами. Разрабатываемая WEB-SCADA доступна на рынке SCADA-систем и представляет собой интеграцию WEB-технологий и систем управления производством в промышленности.

Систему WEB-SCADA разрабатывает и использует российская компания ООО «Элком+» г. Томск, что создаёт особенную ценность для Российского рынка, так как разработка является отечественной. Следовательно, данный программный продукт обладает всеми возможностями для распространения и внедрения в существующую систему управления производством в Российской Федерации.

1 WEB-технологии

Во всем мире сотни миллионов компьютеров, иначе именуемых веб-серверами, подключены к единой сети, называемой Всемирной паутиной. Наиболее распространенная технология передачи данных в этой сети — технология гипертекста.

Веб-технологии — это технологии, используемые во Всемирной паутине. Вся информация в Интернете содержится в форме веб-страницы (минимальная логическая единица). Определенное количество веб-страниц, принадлежащих одному владельцу и расположенных на одном сервере, образуют веб-сайт. Веб-страница и веб-сайт являются основными понятиями, используемыми в веб-технологиях.

Для работы с гипертекстовыми документами используются специальные программы, называемые браузерами. В настоящее время браузеры массово распространяются, и эти программные продукты установлены на персональном компьютере каждого пользователя [1].

Всемирная паутина спровоцировала настоящую революцию в информационных технологиях и дала мощный импульс развитию Интернета.

1.1 Существующие WEB-технологии

Под веб-технологиями определяют весь набор инструментов для организации глобальной сети Интернет. Поскольку в каждом сеансе взаимодействуют две стороны - сервер и клиент, веб-технологии делятся на две группы - технологии на стороне сервера и технологии на стороне клиента.

Технологии на стороне клиента включают все технологии для создания веб-страниц (HTML, JavaScript и т.д.), а технологии на стороне сервера включают технологии для доступа к базам данных в Интернете [2].

В дальнейшем будут рассмотрены наиболее распространённые и изученные в работе WEB- технологии.

1.1.1 HTML и CSS

HTML (от англ. HyperText Markup Language — «язык разметки гипертекста») — стандартный язык разметки для документов во всемирной паутине. Большинство веб-страниц создаются с использованием HTML (или XHTML). Язык HTML интерпретируется браузером и отображается как документ в удобной для пользователя форме.

Текстовые документы, содержащие HTML-код (обычно имеют расширение .html или .htm), обрабатываются специальными приложениями, которые отображают документ в его отформатированном виде. Такие приложения называются «браузерами» и предоставляют пользователю удобный интерфейс для запроса веб-страниц, их просмотра и отправки введенных пользователем данных на сервер. На сегодняшний день самыми популярными браузерами являются Mozilla Firefox, Apple Safari, Google Chrome и Opera [3].

На рисунке 1 представлено взаимодействие между браузером и удалённым веб-сервером с помощью HTML-страниц.

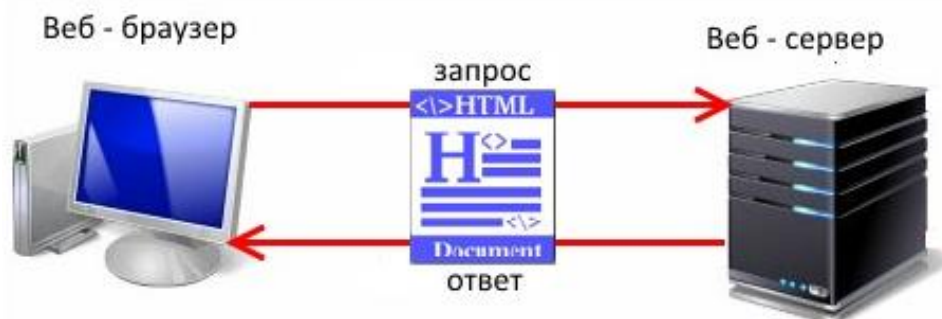


Рисунок 1 – Взаимодействие работы браузеров и веб-серверов с применением HTML-страниц

CSS (англ. Cascading Style Sheets — каскадные таблицы стилей) — технология описания внешнего вида документа, написанного на языке разметки. CSS используется создателями веб-страниц для определения цветов, шрифтов, макета и других аспектов представления документа.

Основной целью разработки CSS было разделение контента (написанного на HTML или другом языке разметки) и представления документа (написанного на CSS). Такое разделение может повысить доступность документа, обеспечить

большую гибкость и возможность управления его представлением, а также снизить сложность и повторяемость структурного содержимого.

Чтобы визуальная часть веб-приложения отображалась перед пользователем, разработчик программного обеспечения сначала создает разметку веб-документа, где присутствуют: заголовок, рабочее пространство, нижняя часть и т. д. После создания разметки разработчик добавляет язык CSS к созданному веб-документу, с помощью которого все размеченные части заполняются цветами, шрифтами и другими компонентами визуальной части [4].

На рисунке 2 представлено схематичное получение HTML-страницы браузером с помощью HTML и CSS.

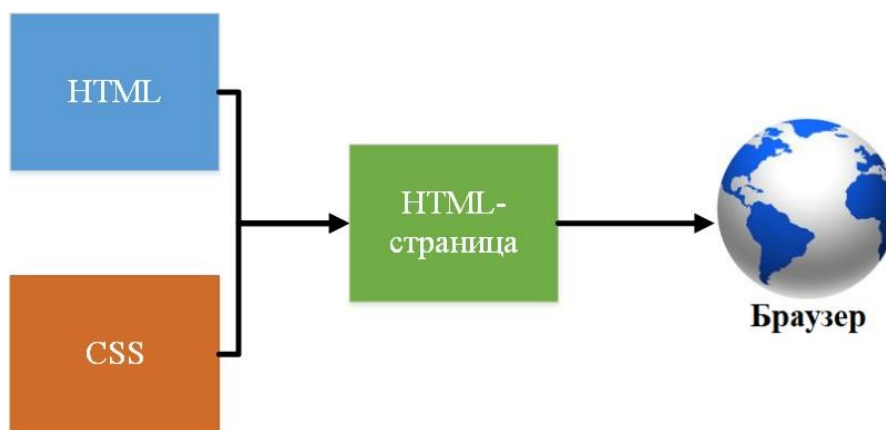


Рисунок 2 – Схематичное получение HTML-страницы браузером с помощью HTML и CSS

1.1.2 JavaScript

Самый простой способ «оживить» веб-страницы, добавить динамические эффекты и настроить реакцию на действия пользователя — это язык сценариев JavaScript. Скрипты (script) JavaScript встроены в веб-страницу или связаны с ней, и после загрузки страницы с сервера они выполняются браузером на стороне клиента. Все современные браузеры имеют поддержку JavaScript.

JavaScript — это компактный объектно-ориентированный язык для создания клиентских веб-приложений. Язык используется для обработки событий, связанных с вводом и выводом информации на веб-страницах.

JavaScript обычно используется в клиентской части веб-приложений, в которых клиент выступает в качестве браузера, а сервер - в качестве веб-сервера, а логика распределена между сервером и клиентом.

В веб-приложениях обмен информацией происходит по сети. Одним из преимуществ этого подхода является независимость клиентов от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами [5].

Клиентская часть среды разработки веб-приложений содержит следующие основные компоненты:

- браузер (browser) - инструмент для просмотра веб-страниц;
- языки для разработки веб-страниц HTML / XHTML и CSS;
- скриптовые языки;
- клиентские расширения, такие как элементы управления ActiveX, Java-апплеты, плагины (плагины, плагины), такие как Macromedia Flash или проигрыватели Silverlight и т. д.

Этот язык используется вместе с HTML-страницами, на которых он работает вместе с логикой, установленной разработчиком: анимация, скрытие или отображение определенных функций веб-приложения [6].

На рисунке 3 представлено схематичное подключение JavaScript к HTML-странице.

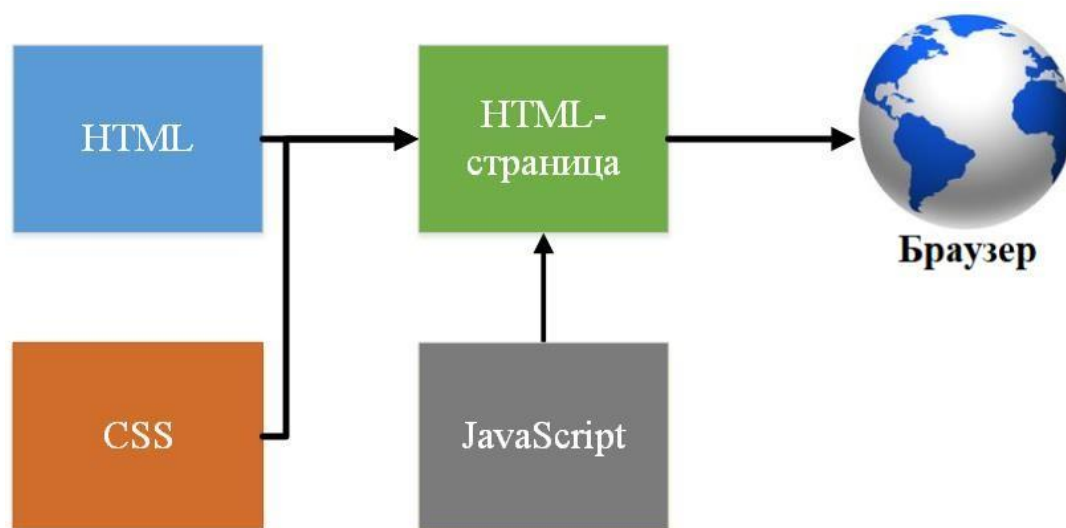


Рисунок 3 – Схематичное подключение JavaScript к HTML-странице

1.1.3 Фреймворк Quasar

Фреймворк (англ. - framework) — это программный продукт, упрощающий создание и сопровождение технически сложных или нагруженных проектов. Как правило, фреймворк содержит только базовые программные модули, а все специфичные для проекта компоненты реализуются разработчиком на их основе. Таким образом, достигается не только высокая скорость разработки, но и высокая производительность и надежность решений.

Преимущества фреймворков:

- повышение скорости и удобства разработки
- сокращение затрат на создание приложения
- оптимизация рабочего времени
- написание чистого кода, не требующего существенного рефакторинга (переработки) в дальнейшем

Недостатки фреймворков:

- Проблемы безопасности. Почти все фреймворки имеют открытый исходный код, что напрямую влияет на их безопасность от внешних угроз.
- Сложность освоения. Фактически, любой фреймворк является результатом работы другого программиста, и не всегда понятно, как он функционирует.
- Необходимость переучивания при смене продукта. К сожалению, в мире не так много фреймворков, которые в идеале похожи друг на друга, поэтому приходится переучиваться [7].

Quasar — это полнофункциональный расширяемый инструмент для разработки современных приложений на основе Vue.js, который позволяет быстро создавать адаптивные веб-сайты и приложения различных типов. Данный фреймворк представляет собой универсальный набор компонентов, которые используются для создания современных приложений. Среди таких компонентов можно отметить, например, диалоговые окна, кнопки, поля ввода, вкладки, панели, всплывающие подсказки, панели инструментов, значки

уведомлений, инструменты для выбора дат, цвета, значки. В то же время Quasar использует как простые компоненты, так и более сложные. Например, это поля для отображения сообщений чата, графики активности, таблицы, календари, видеоплееры, системы меню навигации и многое другое.

Все эти компоненты основаны на Vue.js. Благодаря этому, работа с ними может показаться знакомой разработчикам Vue. Помимо этого, нет сложностей, которые связаны с Webpack и Babel, с алгоритмом tree-shaking и другими способами оптимизации пакетов. Все эти проблемы автоматически решаются внутренними средствами фреймворка [8].

Таким образом, доступ ко всем мощным функциям предназначен для упрощения и ускорения процесса разработки приложений с использованием Quasar.

На рисунке 4 представлен логотип описанного фреймворка.



Рисунок 4 – Логотип фреймворка Quasar

1.1.4 Selenium, WebdriverIO и Mocha

Selenium — это комплексный проект для ряда инструментов и библиотек, которые включают и поддерживают автоматизацию веб-браузеров.

Он предоставляет расширения для эмуляции взаимодействия пользователя с браузерами. В основе Selenium лежит WebDriver – интерфейс для написания наборов инструкций, которые могут быть взаимозаменяемыми во многих браузерах. Возможностями Selenium можно управлять различными

способами, используя множество языков программирования. Именно поэтому Selenium так популярен при разработке автоматизированных тестов [9].

В качестве API для работы с Selenium будет использован WebdriverIO под Node.js. API WebdriverIO обладает удобным синтаксисом, заметно упрощающим разработку тестов. WebdriverIO – гибкая и расширяемая библиотека JavaScript для управления браузерами с сервером Selenium через API WebDriver.

После этого необходима библиотека для описания тестов, в качестве которой будет использоваться Mocha. Mocha — это многофункциональный тестовый фреймворк JavaScript, работающий на Node.js и в браузере, тем самым делая тестирование простым. Тесты Mocha запускаются последовательно, обеспечивая гибкую и точную отчетность.

Mocha обладает двумя достоинствами. Во-первых, Mocha сама по себе является тест-раннером, это избавляет от необходимости использовать дополнительные библиотеки. Во-вторых, достоинством Mocha, как библиотеки для описания тестов, является возможность использования произвольных библиотек ассертов (библиотек, предоставляющих функции сравнения результатов тестов с ожидаемыми) [10].

На рисунке 5 представлены логотипы описанных инструментов.



Рисунок 5 – Логотип инструментов Mocha, Selenium и WebdriverIO

1.2 Промышленный интернет вещей

Для начала необходимо разобраться, чем отличается «Интернет вещей» от «Промышленный интернет вещей».

Интернет Вещей (IoT, Internet of Things) – это система интегрированных

компьютерных сетей и связанных физических объектов со встроенными датчиками и программным обеспечением для сбора и обмена данными, с возможностью удаленного мониторинга и управления в автоматическом режиме (без участия человека).

Промышленный Интернет Вещей (Industrial Internet of Things, IIoT) – интернет вещей для промышленности (отраслевого применения) - система интегрированных компьютерных сетей и связанных промышленных объектов со встроенными датчиками и программным обеспечением для сбора и обмена данными, с возможностью удаленного мониторинга и управления в автоматическом режиме (без участия человека) [11].

Принцип работы технологии Промышленного Интернета Вещей заключается в следующем: изначально датчики, исполнительные механизмы, контроллеры и человеко-машинные интерфейсы устанавливаются на ключевые части оборудования, после чего собирается информация, которая впоследствии позволяет компании получать объективные и точные данные о состоянии предприятия. Обработанные данные доставляются во все отделы предприятия в режиме реального времени, что помогает наладить взаимодействие между сотрудниками различных отделов и принимать обоснованные решения. На рисунке 6 показана концепция промышленного Интернета вещей.



Рисунок 6 – Концепция промышленного интернета вещей

Кроме того, компании могут быстро заменить устаревшие бумажные

документы, а также накопить экспертные знания.

Полученную информацию можно использовать для предотвращения незапланированных простоев, поломок оборудования, сокращения внепланового обслуживания и сбоев в управлении цепочками поставок, что позволяет компании функционировать более эффективно.

При обработке огромного массива неструктурированных данных их фильтрация и адекватная интерпретация являются приоритетом для предприятий. В этом контексте правильное представление информации в удобной для пользователя форме имеет особое значение, для чего в настоящее время на рынке доступны современные аналитические платформы для сбора, хранения и анализа данных о технологических процессах и событиях в режиме реального времени.

Внедрение Интернета вещей требует изменения подходов к созданию и использованию автоматизированных систем управления информацией (АСУ) и общих подходов к управлению предприятиями и организациями. Устаревшие производственные линии, которые по разным причинам не могут быть автоматизированы с помощью IoT, в будущем могут быть заменены новым автоматизированным и роботизированным оборудованием. Еще одним препятствием, ограничивающим развитие IoT, является отсутствие или недостаточно высокое развитие традиционных корпоративных систем управления информацией, вследствие чего решения IoT будут локальными и будут решать нишевые функции и задачи [12].

IoT может последовательно развиваться от соединения отдельных продуктов и объектов (с целью их диагностики и контроля) до объединения различных продуктов и более сложных объектов управления в сети IoT, а сети IoT - в более сложные сетевые платформы и сложные производственные решения.

2 SCADA-системы

На сегодняшний день на многих промышленных предприятиях и в самой отрасли актуальна проблема точного и своевременного управления технологическими процессами и объектами.

Чтобы реализовать решение этой проблемы, мировое сообщество изобрело системы диспетчерского контроля и сбора данных, называемые SCADA-системами.

В настоящее время система SCADA (Supervisory Control And Data Acquisition — диспетчерское управление и сбор данных) — это обязательный атрибут автоматизированной системы диспетчерского контроля. Аппаратной платформой для SCADA-систем являются серверные, а также рабочие станции на базе промышленных компьютеров и операторских панелей [13].

SCADA-системы решают следующие задачи:

- обмен данными в режиме реального времени через специальные драйверы устройств связи;
- алгоритмическое или логическое управление;
- обработка информации в режиме реального времени;
- отображение текущего технологического процесса и другой необходимой информации на экране монитора в понятной и удобной форме для восприятия человеком;
- ведение базы данных в режиме реального времени;
- аварийная сигнализация и её управление;
- подготовка и составление отчётов о ходе технологического процесса;
- сетевое взаимодействия между SCADA ПК и промышленными объектами;
- связь с внешними приложениями (СУБД, электронные таблицы, текстовые процессоры и т. д.);
- анализ и прогнозирование параметров, поступающих с измерительных приборов.

Первые системы SCADA появились в США в 60-х годах прошлого века. Однако наиболее значительное развитие системы произошло в 70-80-х годах XX века благодаря развитию аппаратной элементной базы (микропроцессорной технологии). Основные функциональные возможности SCADA-систем того времени соответствовали возможностям первых управляющих компьютеров. Современные SCADA-системы хорошо структурированы и готовы к использованию [14].

Развитие SCADA-систем — это новый подход к проблемам человеческого фактора в системах управления, направленный в первую очередь на человека (оператора / диспетчера), его задачи и выполняемые им функции.

2.1 Существующие SCADA-системы

В настоящее время на мировом рынке предлагается множество продуктовых решений, реализованных в виде SCADA-систем. Далее будут рассмотрены наиболее популярные на сегодняшний день SCADA-системы.

2.1.1 SCADA «MasterSCADA» компании ИнСАТ (Россия)

SCADA-система «MasterScada» - система для АСУТП, MES, решения задач учета и диспетчеризации, лидер российского рынка.

Основными свойствами программного пакета MasterScada являются масштабируемость, модульность и объектный подход к разработке. Он реализует инструменты и методы для разработки проектов, которые обеспечивают резкое снижение трудозатрат и повышение надежности создаваемой системы. Это первая система в России, где реализуется объектный подход к разработке систем управления процессами.

MasterScada — это полнофункциональный программный пакет SCADA с расширяемой функциональностью. Пакет построен на архитектуре клиент-сервер с возможностью работы как в локальных сетях, так и в Интернете. Помимо работы с верхним уровнем, система позволяет программировать

контроллеры с открытой архитектурой. Прием и передача данных и сообщений на основе стандартов OPC встроены в ядро пакета. Открытые описания интерфейсов и форматов данных, а также максимальная поддержка всех стандартов (XML, HTML, ODBC, OLE, COM / DCOM, ActiveX и т. д.) обеспечивают все необходимые возможности для взаимодействия с внешними программами и системами. На рисунке 7 показан скриншот разработки мнемосхемы в MasterSCADA.

Далее представлены основные преимущества MasterScada для создания систем управления процессами:

- единая среда разработки автоматизированных систем управления;
- отдельная настройка логической структуры объекта;
- открытость и соблюдение стандартов;
- интуитивная простота разработки;
- удобство инструментов;
- удобство методов разработки;
- неограниченная гибкость вычислительных возможностей;
- мощная трехмерная графика и мультимедиа;
- объектный подход [15].

Компании, которые используют данную систему: Роснефть, Газпром, Лукойл, Иркутскэнерго, Калининская АЭС и другие. Общее число внедрений Master SCADA системы составляет десятки тысяч практически во всех отраслях промышленности в России, ближнем и дальнем зарубежье.

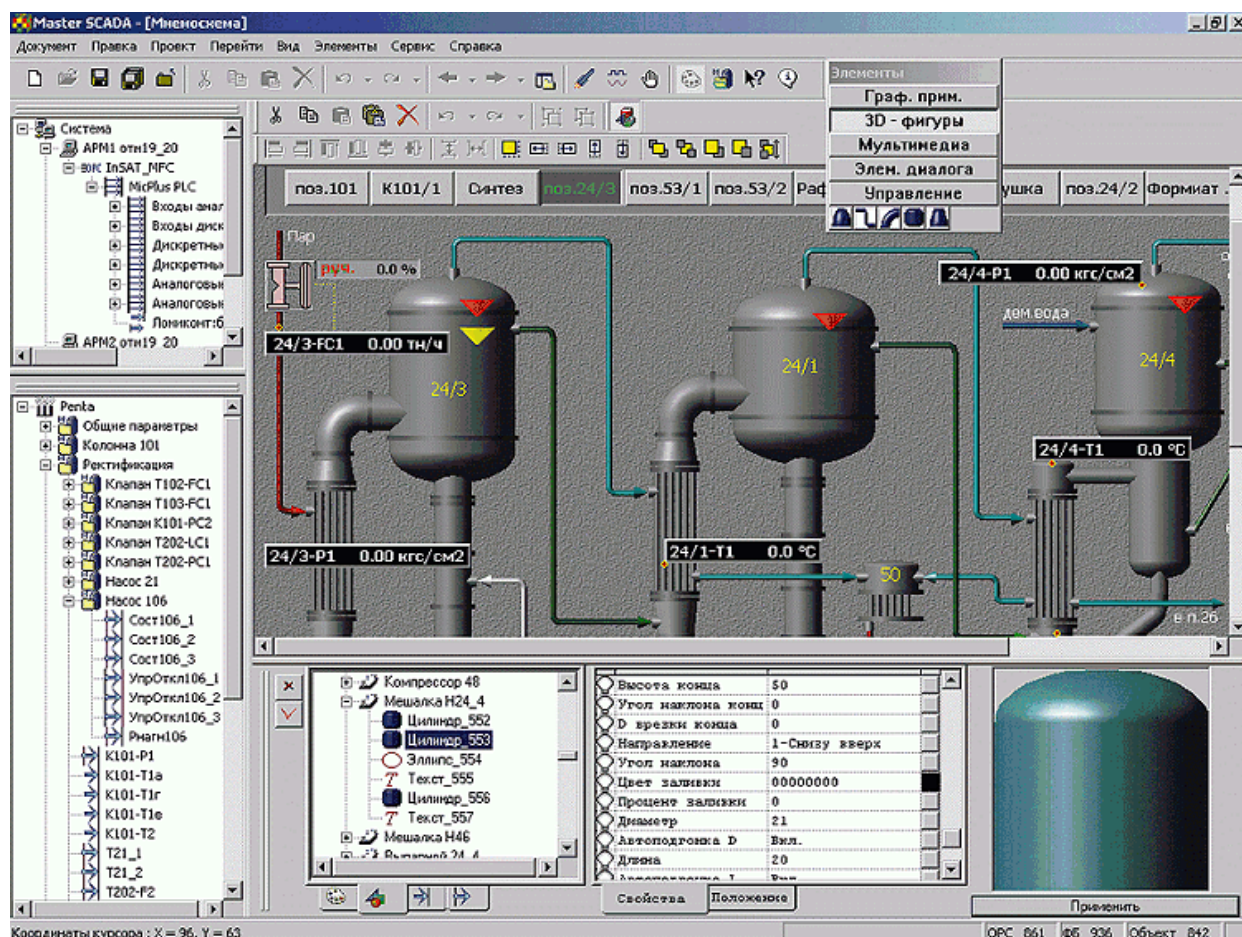


Рисунок 7 – Пример разработки мнемосхемы в SCADA MasterSCADA

2.1.2 SCADA «Trace Mode» компании AdAstra Research Group (Россия)

SCADA «Trace Mode» - программный комплекс класса SCADA HMI, который предназначен для разработки программ для АСУ ТП, систем телемеханики. В настоящее время самая покупаемая в России отечественная программная система для автоматизации технологических процессов.

Система ориентирована на работу с контроллерами и поддерживает все современные информационные технологии. Для подключения контроллеров и сторонних систем можно использовать технологию OPC (Open Platform Communications - открытая коммуникационная платформа). ODBC (Open Database Connectivity - открытая база данных) используется для доступа к базам данных. Драйверы собственной конструкции предусматривают подключение

контроллеров производства ЗАО «Лемикон», ЗАО «Эмикон» и др.

Преимущественные факторы системы Trace Mode:

- все встроенные драйверы (для более чем 2586 ПЛК и УСО) поставляются сразу и бесплатно;
- единый инструмент программирования контроллеров;
- электронная документация и встроенная справочная система;
- масштабируемость от 16 до 1,000,000 точек ввода/вывода;
- средства разработки систем телемеханики;
- самая быстрая система реального времени;
- бесплатный набор драйверов устройств ввода/вывода;
- высокая надежность системы (APM с безударным рестартом);
- демонстрационные проекты TRACE MODE (примеры систем автоматизации, основанные на имитации технологического процесса);
- большая библиотека готовых компонентов (более 1000 шт.);
- адаптивная самонастройка ПИД регуляторов, которая основана на российской запатентованной технологии.

Динамические характеристики и надежность такого программного обеспечения позволяют использовать подготовленные системы автоматизации в таких сферах, как нефтехимия, металлургия, энергетика, машиностроение, а также при проведении исследований.

Программные продукты Trace Mode 6 разделены на интегрированную среду разработки и исполнительные модули. Первая устанавливается на рабочем месте инженера-разработчика и предназначена для создания системы автоматизации. Исполнительные модули предназначены для работы на существующем объекте автоматизации [16].

Особенностью этого программного продукта также является «технология единой линии программирования» — это способность разрабатывать все модули АСУ, используя при этом один инструмент. В рамках одного проекта технология позволяет создать человеко-машинный интерфейс, систему учета ресурсов,

программировать промышленные контроллеры и разработать веб-интерфейс.

На рисунке 8 представлен скриншот разработки мнемосхемы в Trace Mode 6.

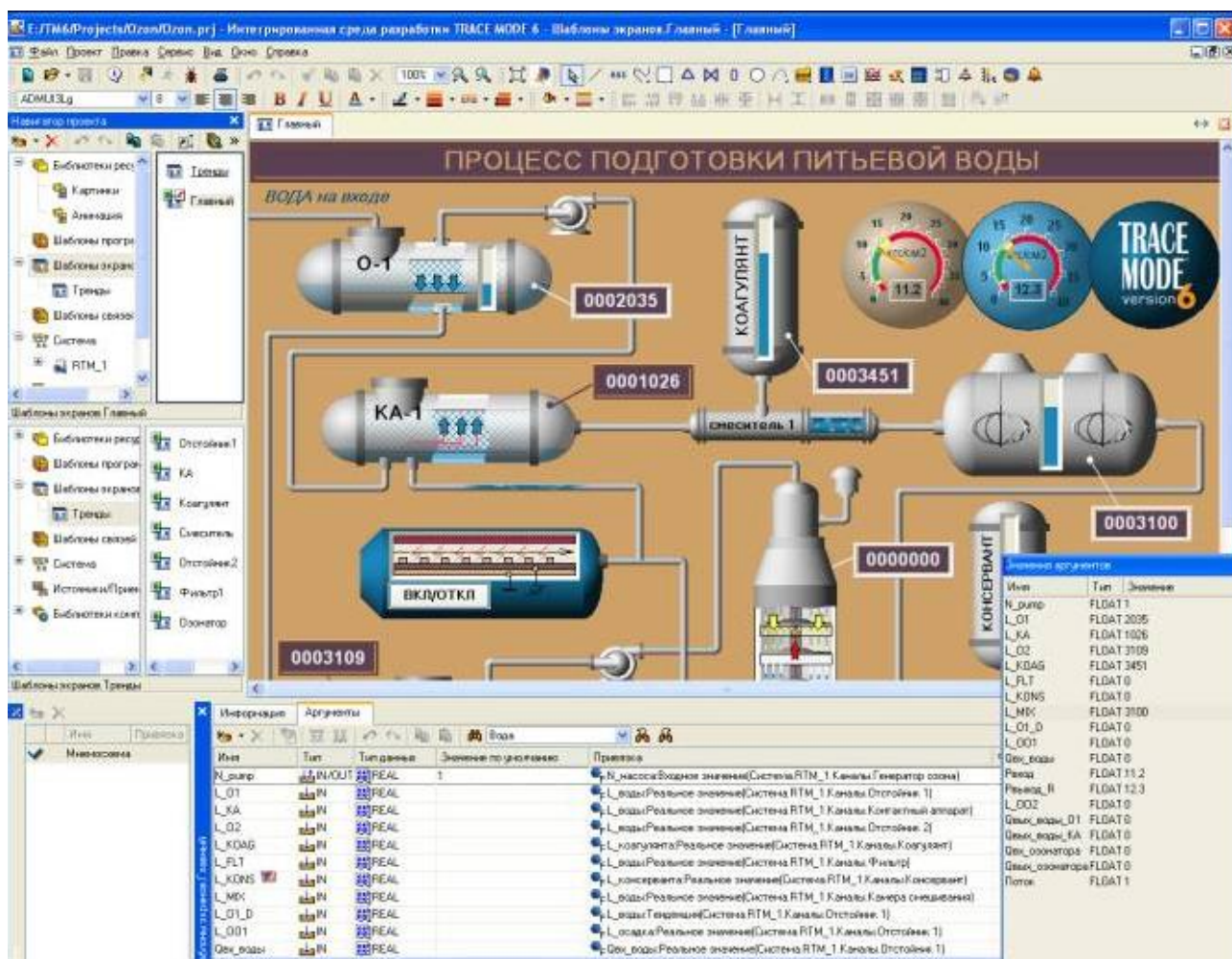


Рисунок 8 – Пример разработки мнемосхемы в SCADA «Trace Mode 6»

2.1.3 SCADA «InTouch» компании Wonderware (США)

Wonderware InTouch HMI – это программное обеспечение для промышленной автоматизации, визуализации и управления технологическими процессами. Данный продукт является одним из лидеров на российском и зарубежном рынках SCADA-продуктов.

Программный пакет InTouch, состоит из двух компонентов – эта среда разработки и среду исполнения. В первой среде создаются мнемонические схемы, определяются и связываются входные и выходные сигналы и параметры,

разрабатываются алгоритмы управления и назначаются права операторов. Такое созданное приложение функционирует в среде выполнения.

Чтобы приложение обменивалось данными с оборудованием, необходимо использовать отдельную программу, которая именуется сервером ввода-вывода. Как правило, такой сервер предназначен для использования с определенным типом оборудования, таким как промышленные контроллеры.

Основные задачи, решаемые с помощью InTouch:

- сбор сигналов (определение состояния процесса производства в текущий момент времени – это температура, давление, положение и т. д.);
- автоматический мониторинг состояния контролируемых параметров, генерация аварийных сигналов и выдача сообщений оператору в графической и текстовой форме;
- отслеживание действия оператора, регистрируя его в системе с помощью имени пользователя и пароля, и назначая ему определенные права доступа;
- снятие контрольных действий на промышленных контроллерах и исполнительные механизмы для непрерывного или дискретного регулирования;
- автоматическая регистрация событий, где записываются изменения производственных параметров, с возможностью просмотра записанных данных в графическом виде, а также регистрации аварий;
- контроль качества продукции путем статистической обработки записанных параметров;
- генерация отчетов [17].

На рисунке 9 представлен скриншот разработки мнемосхемы в InTouch.

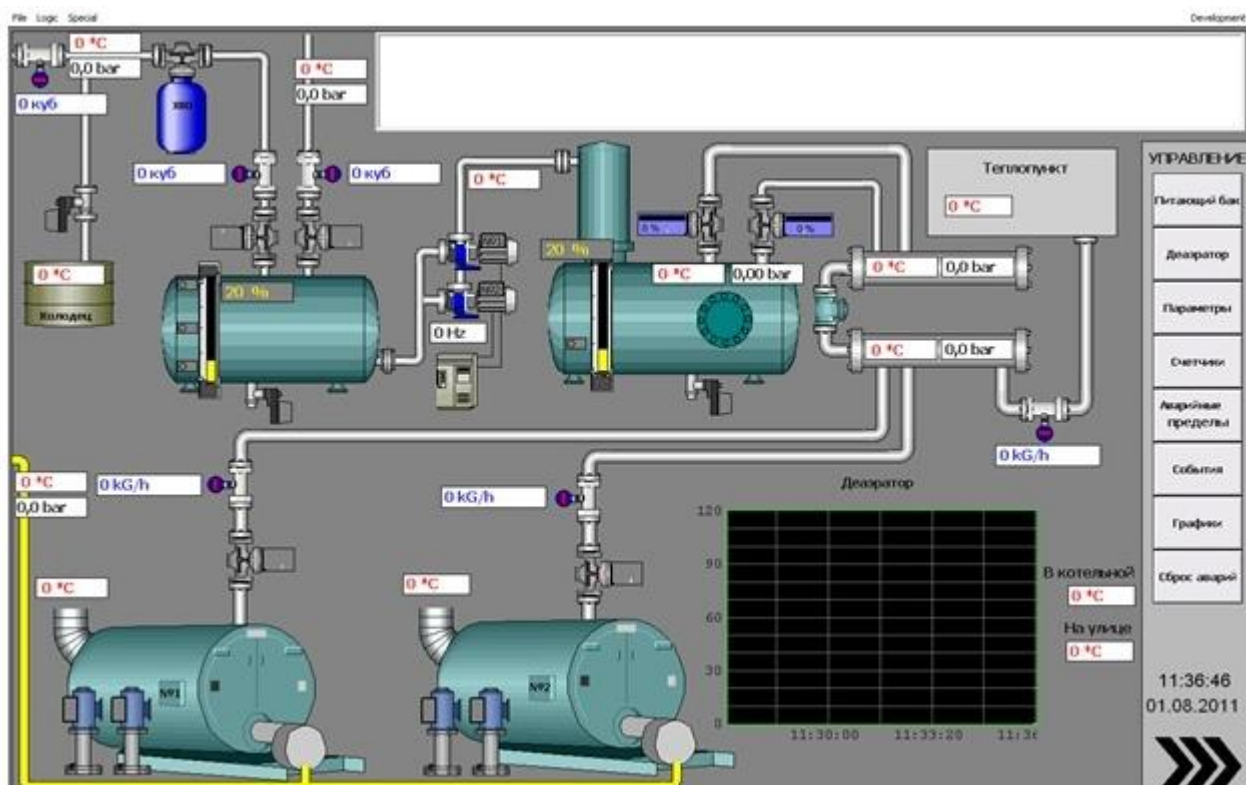


Рисунок 9 – Пример разработки мнемосхемы в SCADA «InTouch»

2.1.4 SCADA «RSView32» компании Rockwell Automation (США)

RSView 32 — это интегрированное программное обеспечение, которое основано на компонентах человеко-машинного интерфейса для мониторинга и управления технологическими процессами. Представленная SCADA-система расширяет возможности обзора процессов благодаря открытым коммуникационным стандартам, обеспечивающим бесперебойное подключение к другим продуктам Rockwell Software, продуктам Microsoft, а также сторонним приложениям.

RSView32 — это первый программный продукт интерфейса человек-машина, который в полной мере использует передовые технологии Microsoft.

Такое преимущество предназначено для возможности:

- открывать графические дисплеи в виде OLE-контейнеров для элементов управления ActiveX;
- разработать объектную модель, чтобы раскрыть часть ее основных

функций для взаимодействия RSVIEW32 с другими программными продуктами на основе компонентов;

- интегрировать Microsoft Visual Basic для приложений (VBA) для безграничной возможности настройки проектов;
- поддерживать стандарты OPC для сервера и клиента для безопасности и быстрой связи с широким спектром оборудования;
- внедрение технологии Add-On Architecture (AOA) для расширения функциональности RSVIEW32 и интеграции новых функций непосредственно в ядро RSVIEW32.

Преимущества использования данной SCADA-системы следующие:

- высокая надежность аппаратного и программного обеспечения;
- интуитивно понятный пользовательский интерфейс;
- использование только открытых стандартов связи;
- совместимость с аппаратными платформами Allen Bradley;
- открытая и гибкая архитектура на основе DNA for Manufacturing Microsoft для производства [18].

На рисунке 10 представлен скриншот разработки мнемосхемы в RSVIEW32.

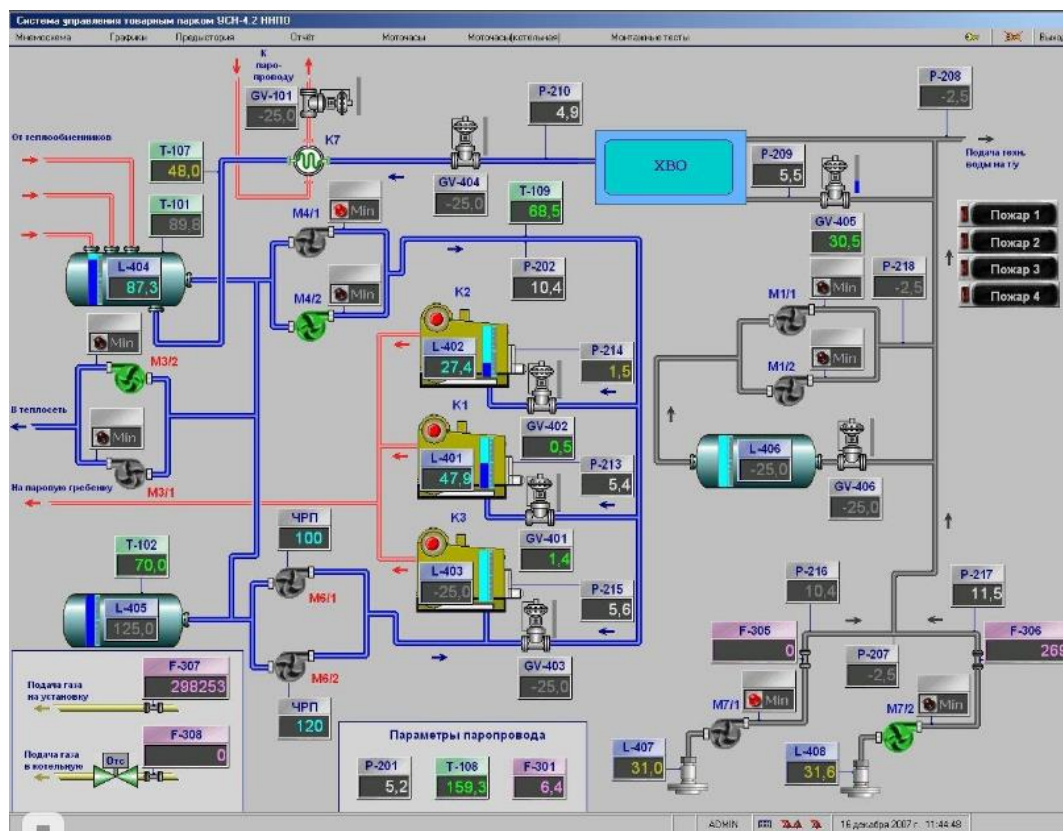


Рисунок 10 – Пример разработки мнемосхемы в SCADA «RSView32»

2.2 Предлагаемое решение

Большинство представленных SCADA-систем хоть и отвечают требованиям настоящего времени, но для перехода в мир WEB им нужны существенные изменения. Однако в связи с реорганизацией структуры взаимодействия программных компонентов между собой усложняется поддержка таких продуктов.

В настоящей магистерской диссертации предлагаемая SCADA-система представляет собой внедрение WEB-технологий в систему автоматизации, что означает возможность удаленного доступа к данным и средствам управления в режиме реального времени в любой момент из любой точки. Некоторые элементы таких систем существовали и раньше, но в связи с появлением WEB-технологий появилось реальное и недорогое решение, которое позволило «сделать видимой» систему АСУ ТП и управлять ею удаленно. Реализация такой системы подразумевает наличие мощной системы защиты информации и

разграничения прав доступа и ролей пользователей, которая уже присутствует во многих современных программных пакетах SCADA.

Преимущество использования WEB-технологий в промышленных системах управления заключается в использовании любого веб-браузера на компьютере диспетчера, независимо от производителя, типа аппаратной платформы или операционной системы.

Большой плюс в управлении и мониторинге (наблюдении) через интернет еще в том, что они могут осуществляться из любой точки земного шара с помощью компьютера или мобильного телефона. Эта возможность особенно важна для высшего руководства.

Другими преимуществами автоматизированных систем с использованием интернета являются:

- снижение затрат на функционирование промышленных систем управления за счет дистанционного управления;
- снижение затрат на обслуживание за счет удаленной диагностики, отладки и обновления программного обеспечения через интернет;
- возможность контролировать состояние производственного или технологического процесса;
- возможность автоматического вызова аварийной службы в случае срабатывания газовых датчиков, дыма, пламени, затопления и т.д.;
- широкий выбор готовых технических решений, аппаратных и программных продуктов для работы с интернетом [19].

3 WEB-SCADA компании ООО «Элком+»

Существует множество требований к любой разрабатываемой автоматизированной системе управления (АСУ), наиболее важным из которых является возможность работы АСУ в режиме реального времени: получать и передавать соответствующие данные.

WEB-SCADA работает на протоколе дуплексной связи WebSocket, который предназначен для обмена сообщениями между браузером и веб-сервером в режиме реального времени и основан на протоколе TCP. Такая технология позволяет только один раз отправлять запрос на сервер, который отдает клиенту измененные данные. В отличие от HTTP, WebSocket позволяет работать с двунаправленным потоком данных, что делает уникальной данную технологию.

Иначе говоря, все данные, которые поступают в браузер WEB-SCADA с серверов АСУ, отображаются в режиме реального времени.

Все технологические данные на промышленном объекте управления поступают с датчиков на устройство под названием AdapTel (адаптер интерфейса для приема и передачи данных телеметрии с распределенных объектов, представленный на рисунке 11), разработанный ООО «Элком+».



Рисунок 11 –Адаптер интерфейса AdapTel

В режиме Transparent Mode устройство передаёт поступающие данные на верхний уровень систем АСУ ТП в сетях стандарта Digital Mobile Radio (DMR) для передачи данных на дальние расстояния с применением радиосвязи. В режиме RTU Mode данные приходят на AdapTel, который передаёт уже пришедшие данные в AdapTel Server, где информация обрабатывается и поступает на WEB-SCADA и СУБД. Передача данных схематично представлена на рисунке 12.

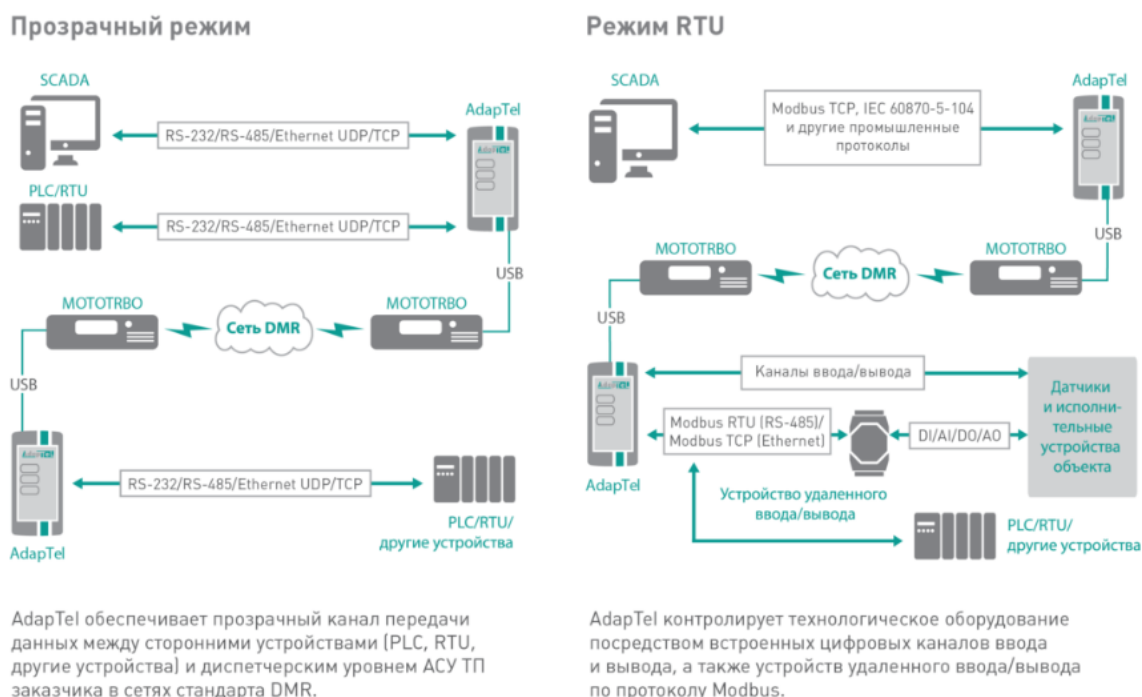


Рисунок 12 – Передача данных в WEB-SCADA

Данные с устройств линейки AdapTel могут поступать не только на WEB-SCADA, но и другие сторонние SCADA-системы без лишнего дублирования информации.

Объектами применения AdapTel могут быть распределённые объекты электроэнергетики, кустовые площадки обустройства объектов нефтедобычи и объекты других промышленных отраслей, где необходим оперативный мониторинг состояния оборудования и передача данных диспетчеру АСУ ТП.

WEB-SCADA обладает всеми функциональными характеристиками SCADA-систем, а также имеет возможность передавать информацию по каналу Smart PTT на программное обеспечение, служащее для организации

диспетчерской радиосвязи на платформе MOTOTRBO [4]. Это даёт возможность представления информации о местоположении аварии по координатам GPS и не только [20].

Далее будет представлен основной интерфейс WEB-SCADA. На рисунке 13 показана страница «Устройства», на которой доступна конфигурация параметров, поступающих с удаленного объекта.

The screenshot displays the 'Demo1 id: Demo1' configuration page. On the left is a sidebar menu with options like 'Неправильные предел', 'Проверка выравнивания', 'Страница с карточкам', 'Event log', 'Trend lines', 'Settings', 'Devices', 'Adaptel_2', 'Adaptel_3', 'Adaptel_4', 'Demo1', 'Demo2', 'Interfaces', 'Users', 'Diagnostics', 'System settings', and 'Licenses'. The main area is divided into two sections: 'Discrete parameters' and 'Analog parameters'. Each section contains a table with various settings.

Discrete parameters table:

Enable	Parameter name	Information	Inversion	Type	Alarm Value	Message on "TRUE"	Message on "FALSE"	Write to DB	On data arrival	On change	On RBE in RTU
<input checked="" type="checkbox"/>	Connect	Состояние связи	<input type="checkbox"/>	Alarm	0	Есть	Нет	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Door	Дверь	<input type="checkbox"/>	Alarm	0	Закрыта	Открыта	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Alarm_system	Охранная система	<input type="checkbox"/>	Alarm	1	Нарушение	Норма	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Sensor	Датчик	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Cmd_Discrete	Команда	<input type="checkbox"/>	Inform	0	Включить	Выключить	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #1	Дискретный вход #1	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #2	Дискретный вход #2	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #3	Дискретный вход #3	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #4	Дискретный вход #4	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #5	Дискретный вход #5	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #6	Дискретный вход #6	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #7	Дискретный вход #7	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Discrete input #8	Дискретный вход #8	<input type="checkbox"/>	Inform	0	Сообщение 1	Сообщение 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Analog parameters table:

Enable	Parameter name	Information	Units	Write to DB	On data arrival	On change	On RBE in RTU	inMin	inMax	outMin	outMax	HiHi	HiHi Message	Hi	Hi Message	Lo
<input checked="" type="checkbox"/>	Voltage	Напряжение	B	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	250	HiHi	240	Hi	210
<input checked="" type="checkbox"/>	Current	Ток	A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	15	HiHi	14	Hi	7
<input checked="" type="checkbox"/>	Temperature	Температура	°C	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	25	HiHi	24	Hi	18
<input checked="" type="checkbox"/>	Humidity	Влажность	%	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	95	HiHi	90	Hi	75
<input checked="" type="checkbox"/>	Analog input #1	Аналоговый вход #1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #2	Аналоговый вход #2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #3	Аналоговый вход #3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #4	Аналоговый вход #4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #5	Аналоговый вход #5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #6	Аналоговый вход #6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50
<input checked="" type="checkbox"/>	Analog input #7	Аналоговый вход #7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 0.1	<input checked="" type="checkbox"/>	0	0	0	0	100	HiHi	75	Hi	50

Рисунок 13 – Интерфейс конфигурации параметров удалённого объекта

Далее на рисунке 14 представлен элемент «Настраиваемая страница», на котором вся информация с датчиков представляет собой вид панели с карточками разного вида: измерительных приборов, аналоговой и дискретной информации, управляющих кнопок и так далее. Интерфейс отображает непосредственно текущее значение контролируемых параметров на объекте.

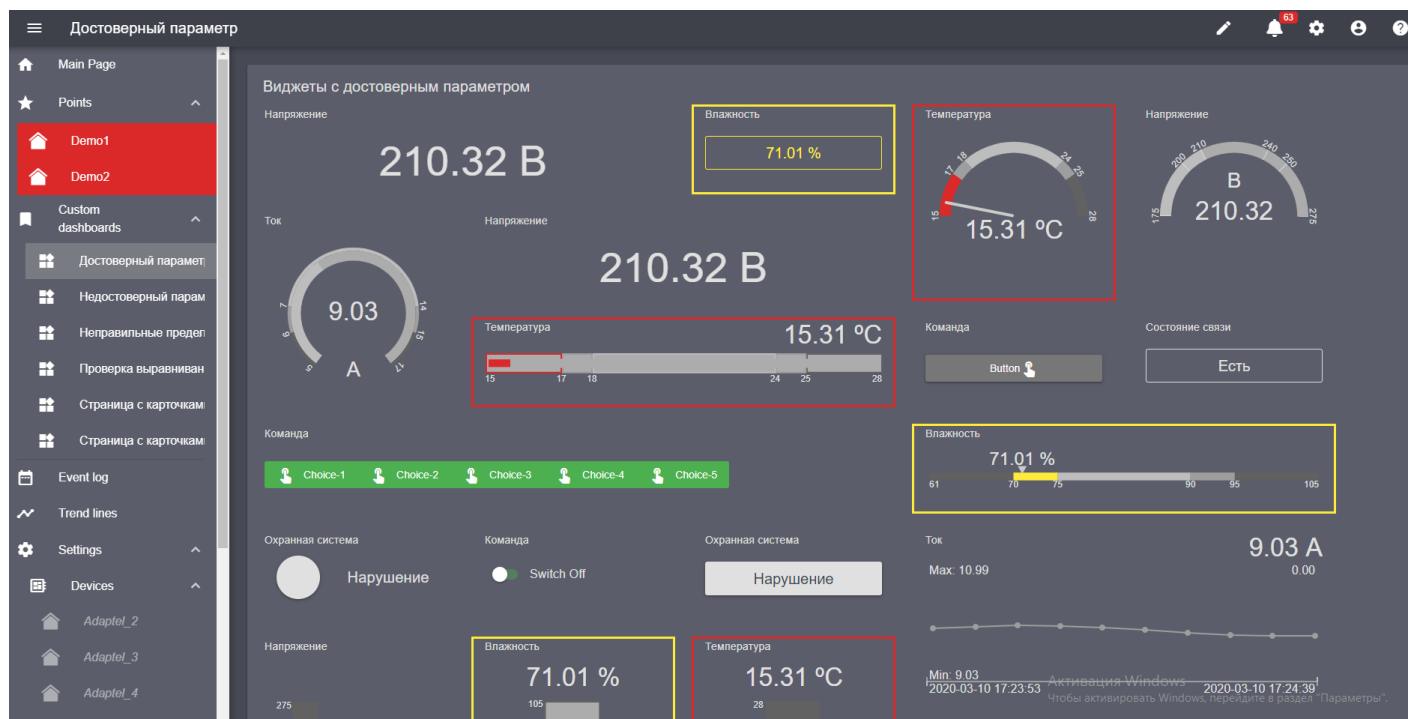


Рисунок 14 – Страница отображения параметров WEB-SCADA

На карточках можно группировать виджеты и настраивать для удобства восприятия. Карточка может содержать неограниченное количество виджетов.

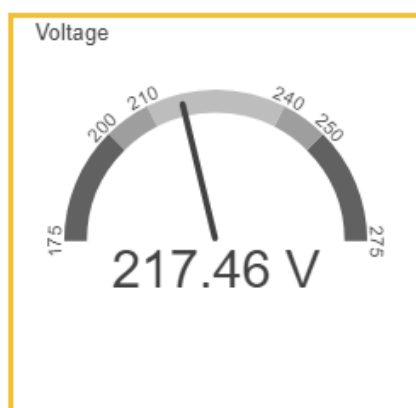
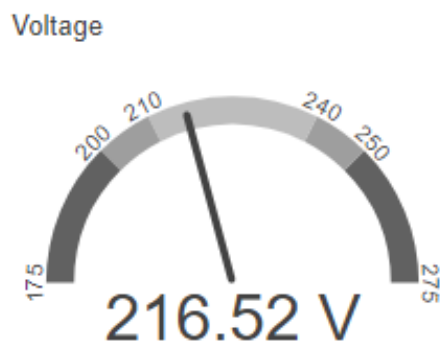
Виджеты в системе делятся на две категории:

а) Виджеты-индикаторы – такие виджеты показывают текущие значения отдельных параметров. В дополнение к текущим значениям они могут отображать предельные значения, пользовательские заголовки и другую информацию в соответствии с настройками.

б) Управляющие виджеты - такие виджеты позволяют изменять значение параметра. Операторы могут выбирать определенные значения или вводить необходимые.

Далее показан пример виджетов-индикаторов:

1) Analog Chip – такой виджет используется для отображения значений в виде рациональных чисел, которые представлены в десятичном виде. На рисунке 15 представлен пример виджета «Analog Chip»



Widget type
Angular Gauge

Parameter
Demo1.Voltage

Show caption

☒ Caption

up to 128 characters

Decimal places

2


 This widget shows the current value of the parameter in the form of gauge. It display the zones, if any.

Рисунок 15 – Виджет типа «Analog Chip» его настройка

2) Bar Gauge Type 1 и Type 2 – такие виджеты отображают аналоговые/численные значения на вертикальной шкале. Настройка, аналогичная предыдущему виджету. На рисунке 16 представлены примеры виджета «Bar Gauge».

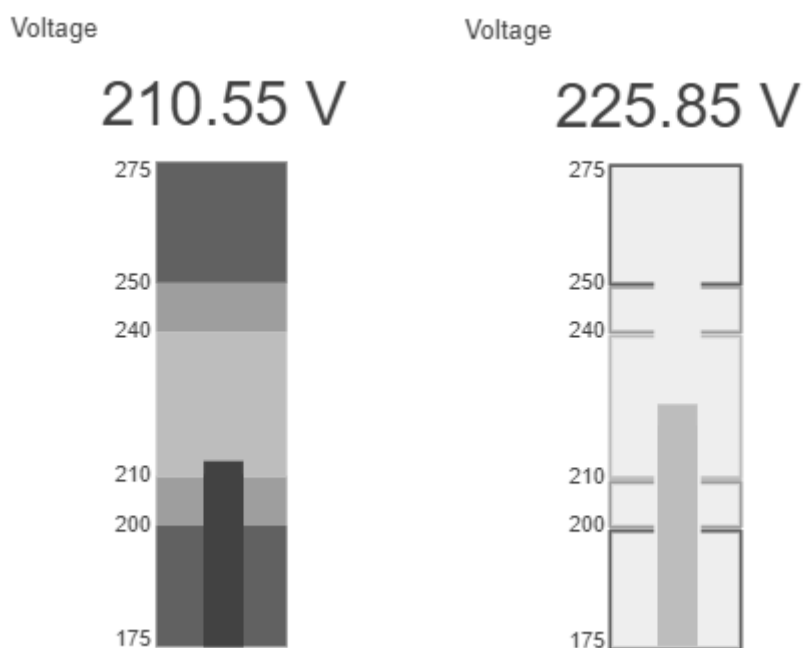


Рисунок 16 – Виджеты типа «Angular Gauge»

3) Discrete Chip – такой виджет показывает текущие значение дискретных логических параметров с использованием цветовой индикации и текста. Цветовая индикация является настраиваемой и должна выбираться по типу параметра (аварийный или информационный). На рисунке 17 показан пример такого виджета.



Рисунок 17 – Виджет «Discrete Chip»

Далее для примера представлены следующие типы управляющих виджетов:

1) Command Button – такой тип виджета устанавливает значение, заданное в настройках виджета. На рисунке 18 представлен пример и настройка виджета «Command Button».

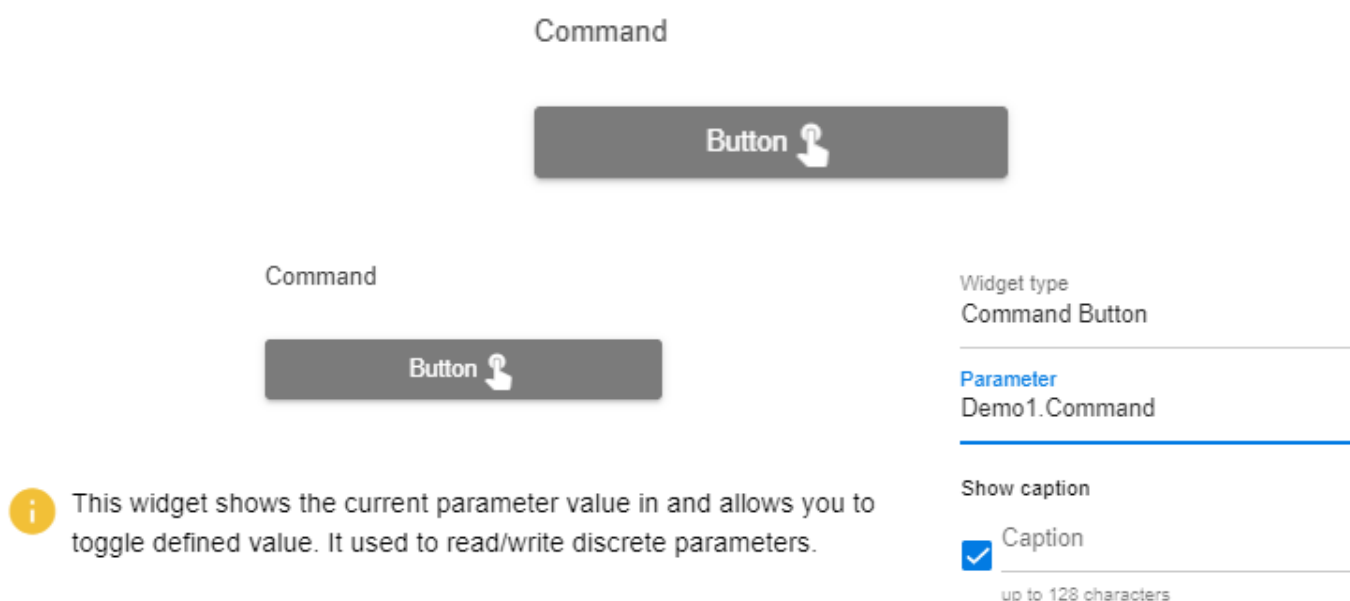


Рисунок 18 – Виджет «Command Button» и его настройка

2) Command Input – такой виджет при нажатии на него открывает диалоговое окно, предлагая ввести требуемое значение. Настройка, аналогичная предыдущему виджету. На рисунке 19 представлен пример виджета «Command Input» [21].

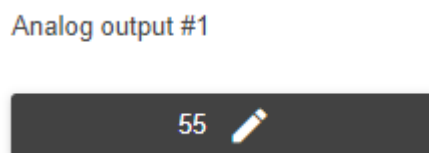


Рисунок 19 – Виджет «Command Input»

Таким образом, продемонстрирована только малая часть типов виджетов для дискретных и аналоговых параметров.

С промышленных объектов технологические данные передаются на Server по технологии Web Socket. Интерфейс Adaptel Server представлен на рисунках 20-22. На рисунке 20 представлен список устройств или объектов, с которых информация поступает на WEB-SCADA.

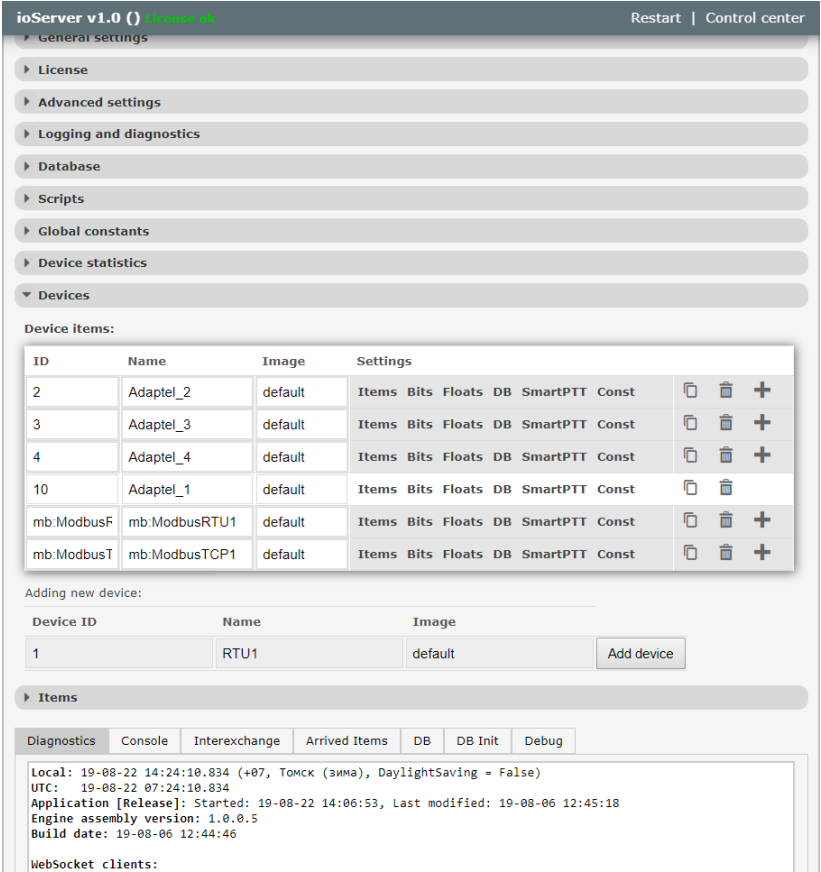


Рисунок 20 – Список устройств в Adaptel Server

На рисунке 21 представлено состояние текущих данных в реальном времени.

F	1	F_0_9 (yTmoJHjkUEyd) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_10 (XMvZWvTPhksK) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_11 (BRegJbvM8Kzt) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_12 (ve5ZB46FEsYm) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_13 (lPGwHb7f53It) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_14 (9M5JOqntOlcI) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_15 (zhq8d55qR0qH) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_16 (j3LIMqd8rMK2) 2019-08-20T11:49:09.384Z	60 60	..	1532 161916.491	
F	1	F_0_17 (F7kEpsiok9bZ) 2019-08-20T11:49:09.512Z 2019-08-22T07:07:15.884Z A HiHi A=2 F=true HiHi	111 111	..	1532 161916.491	ok
F	1	F_0_18 (vTS6MzE2FSom) 2019-08-20T11:49:09.512Z	53 53	..	1532 161916.491	
F	1	F_0_19 (WpdvOhmpY8my) 2019-08-22T16:19:14.512Z 2019-08-22T07:07:15.884Z A Lo A=1 F=true Lo	37.0608 37.0608	..	1532 161916.491	ok
F	1	F_0_20 (Il1LudBvfj52) 2019-08-22T16:19:14.512Z 2019-08-22T07:07:15.884Z A LoLo A=2 F=true LoLo	11.62432 11.62432	..	1532 161916.491	ok
F	1	F_0_21 (VgqU0enWzAys) 2019-08-22T16:19:14.512Z 2019-08-22T07:07:15.884Z A LoLo A=2 F=true LoLo	-91.8784 -91.8784	..	1532 161916.491	ok

Рисунок 21 – Текущие данные в Adaptel Server

На рисунке 22 показан интерфейс настройки каждого параметра для конкретного устройства.

Selected ID: 10 (Adaptel_1 - c0hW0sxbXj7J)

Enable	Type	Category	Name	Cmd	Route	Info			
<input checked="" type="checkbox"/>	Int	▼ Diag	▼ CommStatTX	<input type="checkbox"/>		CommStatTX			
<input checked="" type="checkbox"/>	Int	▼ Diag	▼ CommStatRX	<input type="checkbox"/>		CommStatRX			
<input checked="" type="checkbox"/>	Int	▼ Diag	▼ CommStatSporadic	<input type="checkbox"/>		CommStatSporadic			
<input checked="" type="checkbox"/>	Int	▼ Diag	▼ CommStatCommands	<input type="checkbox"/>		CommStatCommands			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_00	<input checked="" type="checkbox"/>		DO_1_00			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_01	<input checked="" type="checkbox"/>		DO_1_01			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_02	<input checked="" type="checkbox"/>		DO_1_02			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_03	<input checked="" type="checkbox"/>		DO_1_03			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_04	<input checked="" type="checkbox"/>		DO_1_04			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_05	<input checked="" type="checkbox"/>		DO_1_05			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_06	<input checked="" type="checkbox"/>		DO_1_06			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DO_1_07	<input checked="" type="checkbox"/>		DO_1_07			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ B_0_0	<input checked="" type="checkbox"/>		B_0_0			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ B_0_1	<input checked="" type="checkbox"/>		B_0_1			
<input checked="" type="checkbox"/>	Bit	▼ Diag	▼ Started	<input type="checkbox"/>		Started			
<input checked="" type="checkbox"/>	Bit	▼ Diag	▼ Connect	<input type="checkbox"/>		Состояние связи			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_00	<input type="checkbox"/>		DI_1_00			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_01	<input type="checkbox"/>		DI_1_01			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_02	<input type="checkbox"/>		DI_1_02			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_03	<input type="checkbox"/>		DI_1_03			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_04	<input type="checkbox"/>		DI_1_04			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_05	<input type="checkbox"/>		DI_1_05			
<input checked="" type="checkbox"/>	Bit	▼ Field	▼ DI_1_06	<input type="checkbox"/>		DI_1_06			

Рисунок 22 – Редактирование и настройка параметров

Та же самая технология применяется для обратного взаимодействия с объектами управления: пользователь из WEB-SCADA использует кнопки управления для формирования сигналов, которые отправляются через веб-сокеты на сервер для управления устройствами.

4 Тестирование WEB-SCADA

В широком смысле тестирование понимается как метод контроля качества продукта, включающий такие действия, как: планирование работы (Test Management), разработка тестов (Test Design), выполнение тестирования (Test Execution) и анализ результатов (Test Analysis).

Тестирование программного обеспечения (Software testing) — это нахождение ошибок в процессе исследования ПО, а также проверка на соответствие между реальным и ожидаемым поведением программы, проводимая на конечном наборе тестов.

Качество программного обеспечения (Software Quality) — это набор программных функций, связанных со способностью удовлетворять установленные и предполагаемые потребности.

Верификация (verification) — это процесс оценки системы или ее компонентов для определения сопоставления результатов текущей фазы разработки условиям, сформированным в начале этой фазы. Таким образом, происходит проверка выполнения целей, сроков и задач разработки проекта, определенные в начале текущего этапа.

Валидация (validation) — это определение соответствия разработанного программного обеспечения ожиданиям и потребностям пользователя, а также системным требованиям [22].

Цели тестирования:

1. Повысить качество разрабатываемого приложения, которое будет работать правильно при любых обстоятельствах.
2. Повысить вероятность того, что приложение, предназначенное для тестирования, будет соответствовать всем описанным требованиям.
3. Предоставить актуальную информацию о состоянии продукта на настоящий момент.

План тестирования (Test Plan) — это документ, который описывает весь объем работ по тестированию, начиная с описания объекта, стратегии, графика, критериев для начала и окончания тестирования, до оборудования для работы, а

также специальных знаний и оценки рисков с вариантами для их решения.

Разработка тестов (Test design) — это этап процесса тестирования программного обеспечения, на котором сценарии тестирования (тест-кейсы) разрабатываются и создаются в соответствии с ранее определенными критериями качества и целями тестирования.

Тестовый набор (Test Case) — это документация, в которой описывается последовательность действий, конкретных условий и параметров, необходимых для проверки реализации тестовой функции или ее части.

Чек-лист (check list) — это документ, описывающий контрольный список задач, который нужно проверить. В этом случае список может иметь совершенно разные уровни детализации. Насколько подробным будет контрольный список, зависит от требований к отчетности, уровня знания и сложности продукта.

Как правило, чек-лист содержит только действия (шаги), без ожидаемого результата. Контрольный список менее формализован, чем тест-кейс.

Дефект (defect) — это несоответствие между фактическим результатом выполнения программы и ожидаемым результатом. Дефекты обнаруживаются на этапе тестирования ПО, когда тестирующий сравнивает полученные результаты программы с ожидаемым результатом, описанным в требованиях.

Отчет об ошибках (Bug Report) — это документ, описывающий ситуацию или последовательность действий, приводящих к некорректной работе тестового объекта, с указанием причин и ожидаемого результата, т. е. полное описание ошибки [23].

По уровням тестирование бывает:

1. Модульное тестирование (Unit Testing). Тестирование компонентов (модулей) проверяет функциональность и ищет дефекты в частях приложения, которые доступны и могут быть протестированы отдельно (программные модули, объекты, классы и т. д.).

2. Интеграционное тестирование (Integration Testing). Взаимодействие между компонентами системы проверяется после окончания модульного тестирования.

3. Системное тестирование (System Testing). Основной целью такого тестирования является проверка функциональных и нефункциональных требований в системе в целом. В этом случае обнаруживаются дефекты, например, неправильное использование системных ресурсов, неожиданные комбинации данных на уровне пользователя, несовместимость со средой, неожиданные сценарии использования, отсутствующая или неправильная функциональность и т. д.

4. Операционное тестирование (Release Testing). Целью такого тестирования является проверка, что система отвечает всем требованиям и потребностям пользователя и выполняет свою работу в среде эксплуатации согласно бизнес-модели. Операционное тестирование является заключительным этапом валидации.

5. Приемочное тестирование (Acceptance Testing). Комплексный процесс тестирования, который проверяет уровень готовности системы к последующей эксплуатации

Различают 5 направлений при тестировании веб-приложений:

1. Функциональное тестирование (Functional testing);
2. Тестирование удобства пользования (Usability Testing);
3. Тестирование пользовательского интерфейса (GUI Testing);
4. Нагрузочное тестирование (Performance and Load Testing);
5. Тестирование безопасности (Security and Access Control Testing) [24].

По каждому из направлений составляется чек-лист или тест-кейс, в который входит детальное описание того, что именно необходимо протестировать. В данной работе были составлены тест-кейсы в формате таблицы Excel. После составления документации происходит тестирование по созданным тест-кейсам приложения WEB-SCADA. При обнаружении отклонения от данных, описанных в тест-кейсах, дефект фиксируется в серверном веб-приложении для управления проектами и задачами – Redmine.

Далее будет проведено тестирование приложения WEB-SCADA по различным направлениям.

4.1 Функциональное тестирование

Функциональное тестирование – это тестирование ПО в целях проверки реализуемости функциональных требований, то есть способности ПО в определенных условиях решать пользовательские задачи. Функциональные требования определяют действия и решаемые задачи приложения [25].

Так как WEB-SCADA разрабатывается, то функционал тестирования состоял в проверке ПО полноценной SCADA-системы: корректное отображение параметров с датчиков, отображение трендов и их хронология, корректное ведение логирования событий, их архивирование и многие другие аспекты SCADA-систем.

Для осуществления функционального тестирования были написаны автоматизированные тест-кейсы с помощью ПО Browser Automation Studio [6], которое позволяет создавать программы для автоматизации действий в WEB-сфере.

На рисунке 23 показан интерфейс программы BAS.

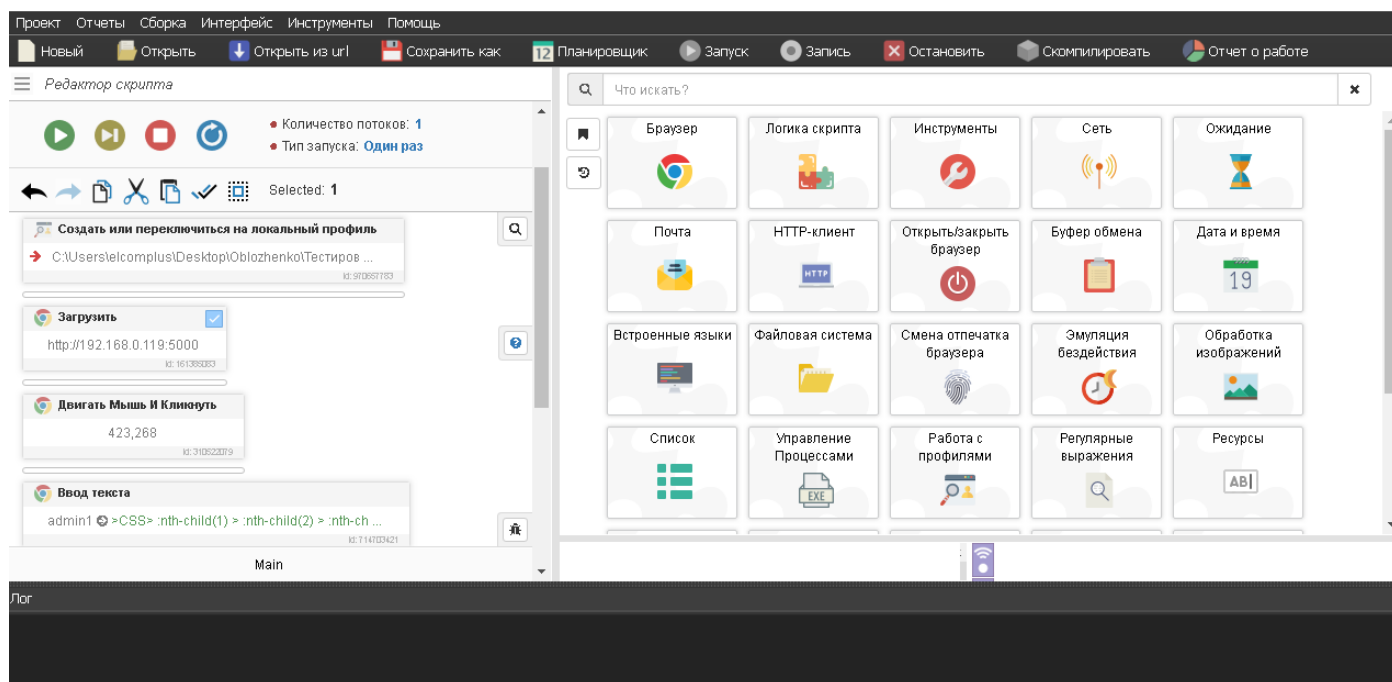


Рисунок 23 – Интерфейс программы Browser Automation Studio

Тест-кейс функционального тестирования был составлен в формате таблицы Excel, пример которого представлен в таблице 1.

Таблица 1 – Пример тест-кейса функционального тестирования

Предусловие: - SCADA версии SmartPTT SCADA 1.0.1.192 или выше и установлена лицензия для SCADA - Одно устройство добавлено в систему и сконфигурировано (Adaptel) - Язык интерфейса Русский Путь до файлов с изображениями: \\fileserver\ASU\10 Проработка решений\Web-SCADA (Система мониторинга)\8_Testing\Backup\Pictures		
№ п/п	Шаги (Вид и описание выполняемых операций по тестированию)	Ожидаемый результат
3.1	Для диспетчера: запустить Диспетчер SmartPTT	Откроется диалоговое окно для входа в систему
3.2	В строке Оператор выбрать Administrator Строку Пароль оставить пустым Нажать кнопку "Ок"	Произойдёт вход в систему на страницу "Главная" под учётной записью администратора системы Загрузится интерфейс системы
3.3	Для автономной SCADA: в адресной строке браузера ввести: https://localhost:5001	Откроется окно авторизации, где необходимо ввести Имя и пароль созданного пользователя и авторизоваться
3.4	Нажать на кнопку "Login"	
3.5	Откроется WEB-SCADA	Произойдёт вход в систему на страницу "Главная" под учётной записью администратора системы Загрузится интерфейс системы
3.6	В левом раскрываемом меню SCADA навести указатель мыши на элемент "Настраиваемые страницы" и нажать на иконку «+», чтобы создать страницу.	Произойдёт переход на новую создаваемую страницу Содержание страницы является пустым Страница в режиме редактирования
3.7	Изменить наименование страницы, нажать на кнопку Сохранить	Название страницы изменится на новое, режим редактирования закроется
3.8	В верхнем меню нажать "Редактировать эту страницу"	На верхней панели появятся тултипы для кнопок "Масштаб страницы", "Добавить карточку на страницу", "Удалить эту страницу" и "Отменить редактирование этой страницы"
3.9	Нажать на кнопку "Добавить карточку на страницу", расположенной в верхнем меню	Появится раскрывающийся список с доступными элементами для добавления
3.10	Нажать на элемент "Добавить карточку-картинку" раскрытого списка	На странице появится новая пустая карточка с управляющими элементами (тултипы: переместить карточку, выбрать файл, добавить виджет на карточку (заблокировано), редактировать карточку, удалить карточку), а также координаты (X, Y), ширина и высота, редактирование и удаление виджета
3.11	Нажать на управляющий элемент редактировать карточку: - Выбрать максимальную ширину. - Нажать кнопку Закрыть	Ширина карточки займёт всю страницу
3.12	Навести указатель мыши на управляющий элемент "Добавить виджет на карточку"	Стрелка указателя мыши превратится в знак блокировки, символизирующий о невозможности добавления виджета на карточку, пока не будет загружено изображение в качестве фона
3.13	Нажать на кнопку "Выбрать файл" для добавления изображения на карточку, представленный в виде иконки изображения	Откроется окно проводника для выбора файла

3.14	Из папки с файлами изображений выбрать файл 111.jpg. Нажать "Открыть"	Окно выбора файла закроется. На карточке появится выбранное изображение
3.15	Нажать на управляющий элемент "Добавить виджет на карточку"	Появляется окно с выбором типа виджета и параметра В нижней части окна расположены кнопки: - Закрыть - Принять
3.16	Нажать за пределы области появившегося окна редактирования виджета	Окно редактирования виджета останется на месте (не закроется)
3.17	Навести курсор мыши на кнопку "Принять"	Кнопка заблокирована
3.18	Нажать на кнопку "Закрыть"	Окно редактирования виджета закроется
3.19	Реализовать следующие варианты действий: 1) - Нажать на управляющий элемент "Добавить виджет на карточку" - Выбрать произвольный тип виджета - Навести указатель мыши на кнопку "Принять" - Закрыть окно 2) - Нажать на управляющий элемент "Добавить виджет на карточку" - Выбрать произвольный параметр - Навести указатель мыши на кнопку "Принять" - Закрыть окно	1) Кнопка заблокирована 2) Стрелка указателя мыши не блокируется (тип виджета и параметр выбраны)
3.20	Для каждого из типов виджетов: - Quarter gauge - Bullet gauge - Bar gauge - Rectangle - Value - Ellipse реализовать следующие действия: 1) Нажать на управляющий элемент "Добавить виджет на карточку" 2) Выбрать соответствующий параметр во вкладке "Параметр" 3) Нажать на кнопку "Принять"	Добавленные виджеты размещены на карточке поверх картинки Вокруг виджета отображена граница в виде пунктирных линий Управляющие элементы виджета изображены на верхней панели карточки в виде иконок: редактирование виджета и его удаление, а также координаты XY, Ширина и Высота

Были созданы такие тест-кейсы, как: проверка создания backup / сохранение backup / загрузка backup; проверка реализации карточки Image; проверка функциональных возможностей исторического журнала и исторического графика; проверка функциональных возможностей виджетов на карточке; проверка квитирования аварий; редактирование имени и прав пользователей и т. д.

Помимо этого, производилось дополнение и редактирование тест-кейсов по WEB-SCADA. К примеру, были дополнены такие тест-кейсы, как: проверка OPC-клиента; проверка подключения к ioServer; проверка настройки для ModbusMaster: протокол Modbus TCP и протокол Modbus RTU; проверка ограничения лицензии по функциям и др.

4.2 Юзабилити-тестирование

Юзабилити-тестирование – это тестирование удобства интерфейса продукта для его предполагаемого применения. Такой тип тестирования показывает, насколько продукт соответствует ожиданиям пользователей, выявляет проблемные места в интерфейсе, дает возможность взглянуть на продукт глазами пользователей. Такое тестирование позволяет определить эргономичность (приспособленность к использованию) программы.

При юзабилити-тестировании рассматривались следующие основные вопросы:

- Понятен ли интерфейс WEB-SCADA пользователям?
- Удобен ли интерфейс?
- Удобная навигация?
- Есть ли интуитивно-непонятные моменты в работе с WEB-SCADA?
- Какое впечатление может создаться у пользователей по мере работы с приложением?
- Есть ли что-то лишние или что можно добавить?

Тестирование удобства пользования может проводиться на разных этапах разработки продукта. Однако такое тестирование рекомендуется проводить уже

на начальных этапах разработки интерфейса. Это позволит сразу внести необходимые корректировки и сделать интерфейс удобным, тем самым потратить на разработку меньше времени и средств [26].

Чек-лист юзабилити-тестирования представлен в таблице 2.

Таблица 2 – Пример чек-листа юзабилити-тестирования

№	Вид и описание выполняемых операций по тестированию	Результаты
п/п		
1	Main	
1.1.	Понятливость таблицы (что отображает)	+
1.2.	Понятливость характеристик записей	+
1.3.	Кнопки навигации по таблице	+
1.4.	Удобный поисковик	+
2	Devices dashboards	
2.1.	Обновление страницы	~
2.2.	<i>Карточки</i>	
2.2.1.	Добавление	+
2.2.2.	Редактирование	+
2.2.3.	Удаление	~
2.2.4.	Изменение размера	+
2.2.5.	Перемещение	+
2.2.6.	Параметры	
а)	Добавление параметра	+
б)	Удаление параметра	+
в)	Редактирование параметра	+
2.3.	Редактирование панели	+
2.4.	Удаление панели	~
3	Log -> Historical alarms & events	
3.1.	Сортировка	+
3.2.	Поисковик	+
3.3.	Обновление страницы	+
3.4.	Разворот страницы на весь экран	+
4	Settings -> Devices	
4.1.	Обновление страницы	+
4.2.	Панель навигации устройств	+
4.3.	Подсветка строки в таблице при наведении указателя мыши	+
4.5.	Панель навигации в таблицах	+
4.6.	Редактирование параметров	+
4.7.	Поисковик	+
4.8.	Редактирование параметров	

4.3 Тестирование интерфейса

Тестирование пользовательского интерфейса проверяет, насколько ожидаемо ведет себя программа и как элементы интерфейса отображаются на различных устройствах, когда пользователь выполняет определенные операции. Это позволяет оценить, насколько эффективна работа пользователя с приложением и каким образом элементы интерфейса реагируют на действия пользователя. Такое направление тестирования позволяет убедиться, что все функциональные взаимодействия между веб-сервером и интерфейсом пользователя воспроизводятся в соответствии с такими требованиями, как:

- функционирование приложения с различными размерами экранов;
- кроссбраузерность;
- единый стиль Material Design.

Также с помощью него проверяется корректность отображения сообщений об ошибке.

При тестировании учитываются такие критерии, как:

- минимальное время для выполнения задач пользователем;
- минимальное количество ошибок, которые пользователь допускает при работе с приложением;
- полное понимание интерфейса пользователями и отсутствие неясностей при работе с ним;
- минимальный объем информации, вводимой пользователями;
- простота и визуальная привлекательность интерфейса.

Для тестирования пользовательского интерфейса проводится кросс-браузерное и многоплатформенное тестирование. Ожидаемый результат – высококачественное приложение, которое работает на всех типах устройств и во всех современных браузерах [27].

В таблице 3 приведен пример чек-листа тестирования интерфейса.

Таблица 3 – Пример чек-листа тестирования интерфейса

№	Вид и описание выполняемых операций по тестированию	Результаты
п/п		
1	Провекра работы приложения с различными размерами экранов	
1.1.	1366x768	+
1.2.	360x640	+
1.3.	1920x1080	/-
1.4.	375x667	+
2	Main	
2.1.	Выравнивание таблицы	/-
2.2.	Корректное отображение букв / символов	+
3	Проверка кроссбраузерности	+
3.1.	Google Chrome	+
3.2.	Mozilla Firefox	+
3.3.	Opera	/-
3.4.	Yandex	+
4	Devices dashboards	
4.1.	Карточки	
4.1.1.	Отображение числовых параметров	~
4.1.2.	Действия с карточками	
<i>4.1.2.1.</i>	<i>Добавление карточки и сохранение изменений на доске</i>	+
<i>4.1.2.2.</i>	<i>Добавление карточки и отмена изменения сохранения доски</i>	+
<i>4.1.2.3.</i>	<i>Добавление параметров на карточку</i>	+
<i>4.1.2.4.</i>	<i>Удаление параметров с карточки</i>	+
4.2.	Панели	
4.2.1.	Функциональная кнопка добавления карточки на панель	+
4.2.2.	Отображение карточек	+
4.2.3.	Отображение поля наименования панели	
4.2.4.	Отображение числовых параметров	+
4.2.5.	Отображение числовых параметров	/-
4.2.6.	Функциональная кнопка редактирования панели	/-
4.2.7.	Функциональная кнопка сохранения панели	

4.4 Нагрузочное тестирование

Этот тип тестирования также называется тестированием производительности и необходим для моделирования одновременной работы определенного количества пользователей. Нагрузочное тестирование сайта проверяется в большинстве случаев автоматом, то есть специальными программами.

Данный вид тестирования производился с помощью программного обеспечения Apache JMeter. ПО позволяет эмулировать заданное количество пользователей, которые могут выполнять заданные действия с веб-приложением.

На рисунке 24 представлен интерфейс программы Apache JMeter, позволяющий создавать большое количество запросов, используя несколько компьютеров при управлении этим процессом с одного из них. Архитектура, которая поддерживает сторонние плагины, позволяет добавлять новые функции в инструмент.

В программе реализована авторизация для виртуальных пользователей, а также поддерживаются пользовательские сессии. Присутствует регистрация результатов испытаний и различная визуализация результатов в виде диаграмм, таблиц и т. п. [28].

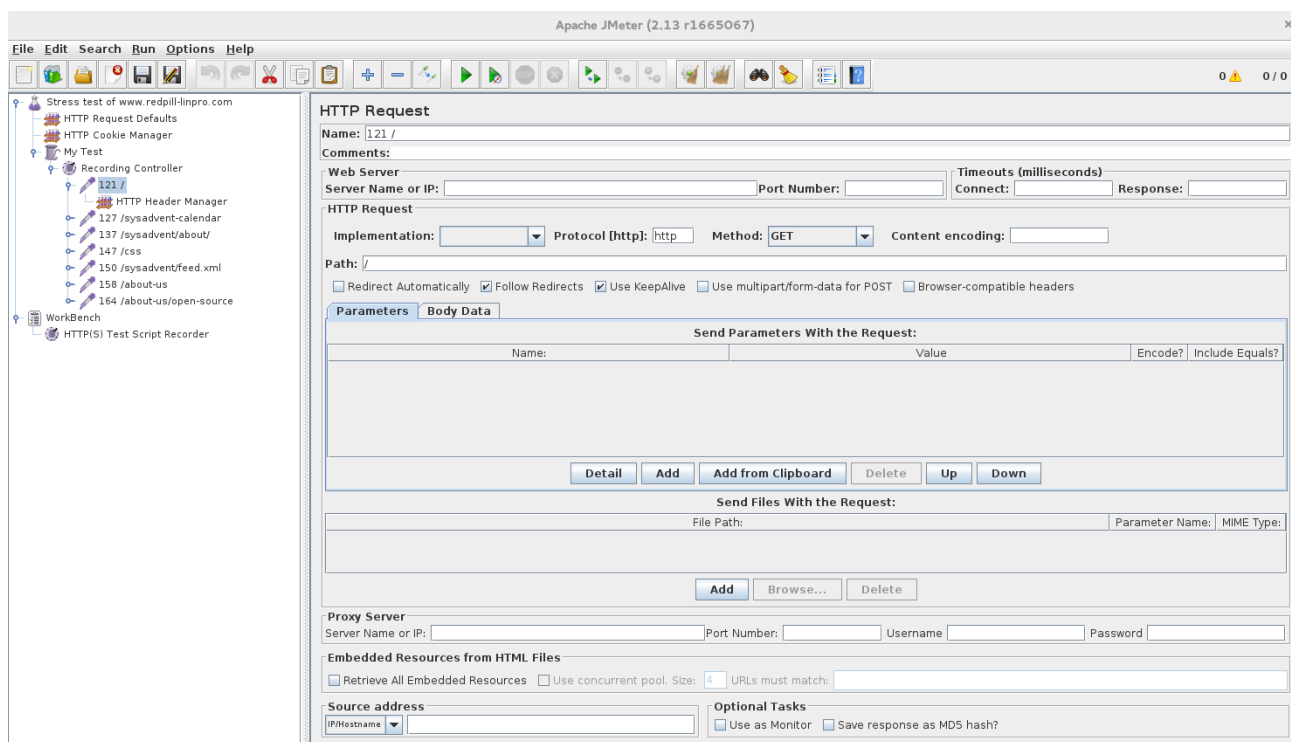


Рисунок 24 – Интерфейс Apache JMeter

Добавим в Thread Group элементы представления по очереди. Кликаем правой клавишей на Thread Group — Add — Listener — Graph Results, View Result in Table и др. Здесь необходимо выполнить все необходимые настройки (в том числе настройка HTTP Proxy Server).

Последняя настройка перед запуском — необходимо указать количество пользователей (Number of Threads) и количество итераций (Loop Count). После запуска теста можно посмотреть онлайн-статистику (рис. 25-26).

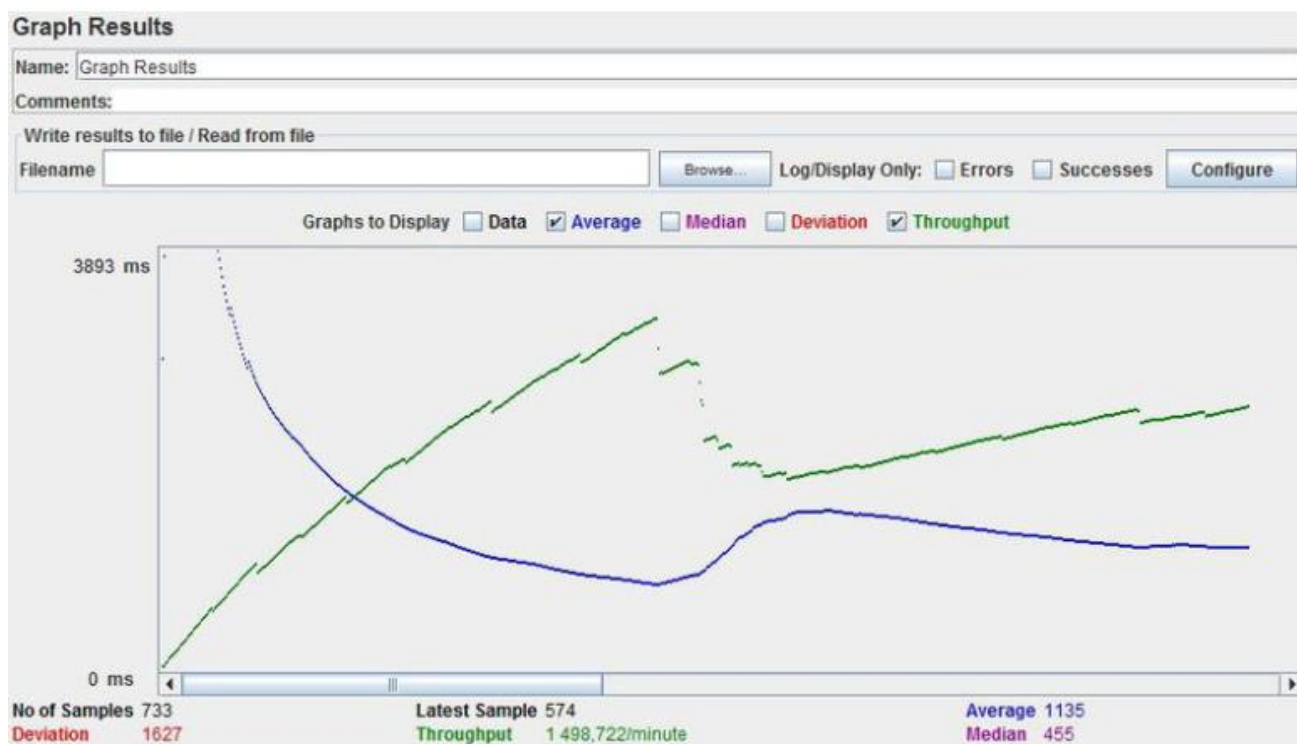


Рисунок 25 – Элемент просмотра Graph Results

Значения приведены в миллисекунда, где:

- Data (данные) — время отклика каждого отдельного блока;
- Average (среднее) — среднее время отклика, объективный график изменения нагрузки;
- Median (медиана) — значение медианы;
- Deviation (отклонение) — ошибка, стандартное отклонение;
- Throughput (пропускная способность) — пропускная способность выполняемых запросов.

На рисунке 26 показано, что время отклика значительно увеличивается: чем выше пропускная способность запросов (меньше миллисекунд), тем больше времени требуется серверу для обработки.

Далее рассмотрим статистику View Result in Table (рис. 26)

View Results in Table							
Name: View Results in Table							
Comments:							
Write results to file / Read from file							
Filename		Browse...	Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes		Configure		
Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Latency
687	12-23:56.518	Thread Group 1-15	/pagemethods.a...	2239		6303	2239
688	12-23:58.289	Thread Group 1-8	/Constants.ashx	469		782	469
689	12-23:56.580	Thread Group 1-18	/pagemethods.a...	2180		6303	2180
690	12-23:58.368	Thread Group 1-22	/pagemethods.a...	416		655	416
691	12-23:58.387	Thread Group 1-26	/pagemethods.a...	413		655	413
692	12-23:58.548	Thread Group 1-23	/pagemethods.a...	262		655	262
693	12-23:58.560	Thread Group 1-25	/pagemethods.a...	255		654	255
694	12-23:58.569	Thread Group 1-16	/pagemethods.a...	249		655	249
695	12-23:58.604	Thread Group 1-20	/pagemethods.a...	218		655	218
696	12-23:58.592	Thread Group 1-9	/pagemethods.a...	239		655	239
697	12-23:58.590	Thread Group 1-17	/pagemethods.a...	242		655	242
698	12-23:58.589	Thread Group 1-5	/pagemethods.a...	244		655	244
699	12-23:58.605	Thread Group 1-28	/pagemethods.a...	228		655	228
700	12-23:58.591	Thread Group 1-11	/pagemethods.a...	242		655	242
701	12-23:58.670	Thread Group 1-21	/pagemethods.a...	163		655	163
702	12-23:58.683	Thread Group 1-7	/pagemethods.a...	152		655	152
703	12-23:58.783	Thread Group 1-8	/pagemethods.a...	152		655	152
704	12-23:58.368	Thread Group 1-14	/pagemethods.a...	1026		2264	0

Рисунок 26 – Элемент просмотра View Result in Table

На рисунке отображается результат в виде таблицы, где указано время и статус (успешно/не успешно). Таким образом, с помощью Apache JMeter было проведено нагрузочное тестирование для ПО WEB-SCADA и получены наглядные результаты.

4.5 Тестирование безопасности

Под безопасностью любого веб-приложения понимается принцип соблюдения трёх основ безопасности: конфиденциальность, целостность информации и доступность.

Конфиденциальность — это комплекс мер по предотвращению утечки и разглашения той или иной информации. Сюда также входит ограничение доступа к ресурсу определенной категории пользователей.

Целостность — это обеспечение достоверности и полноты информации; ожидание, что ресурс будет изменен только соответствующим образом определенной группой пользователей.

Доступность — это требование, по которому ресурсы доступны для авторизованного пользователя, внутреннего объекта или устройства.

Существуют специальные программы, при помощи которых можно

производить комплексное тестирование веб-приложений.

В данном случае была применена программа Acunetix Trial (рисунок 27), которая производит сканирование веб-приложений по следующим уязвимостям:

- тестирование приложения к различному виду инъекций (SQL, SOAP, LDAP, XPATH и т.д.);
- тестирование приложения к XSS-уязвимостям;
- проверка HTTP заголовков;
- проверка редиректов и переадресаций;
- и многое другие.

Acunetix предлагает интерактивное тестирование безопасности приложений (IAST) с помощью технологии тестирования уязвимостей AcuSensor «серый ящик». Сканер предлагает обнаружение уязвимостей наивысшего уровня и сканирует более 1200 известных уязвимостей в ядре, темах и плагинах WordPress. Технология DeepScan Acunetix позволяет сканировать все веб-сайты, в том числе разработанные с использованием HTML5, JavaScript и RESTful API [29].

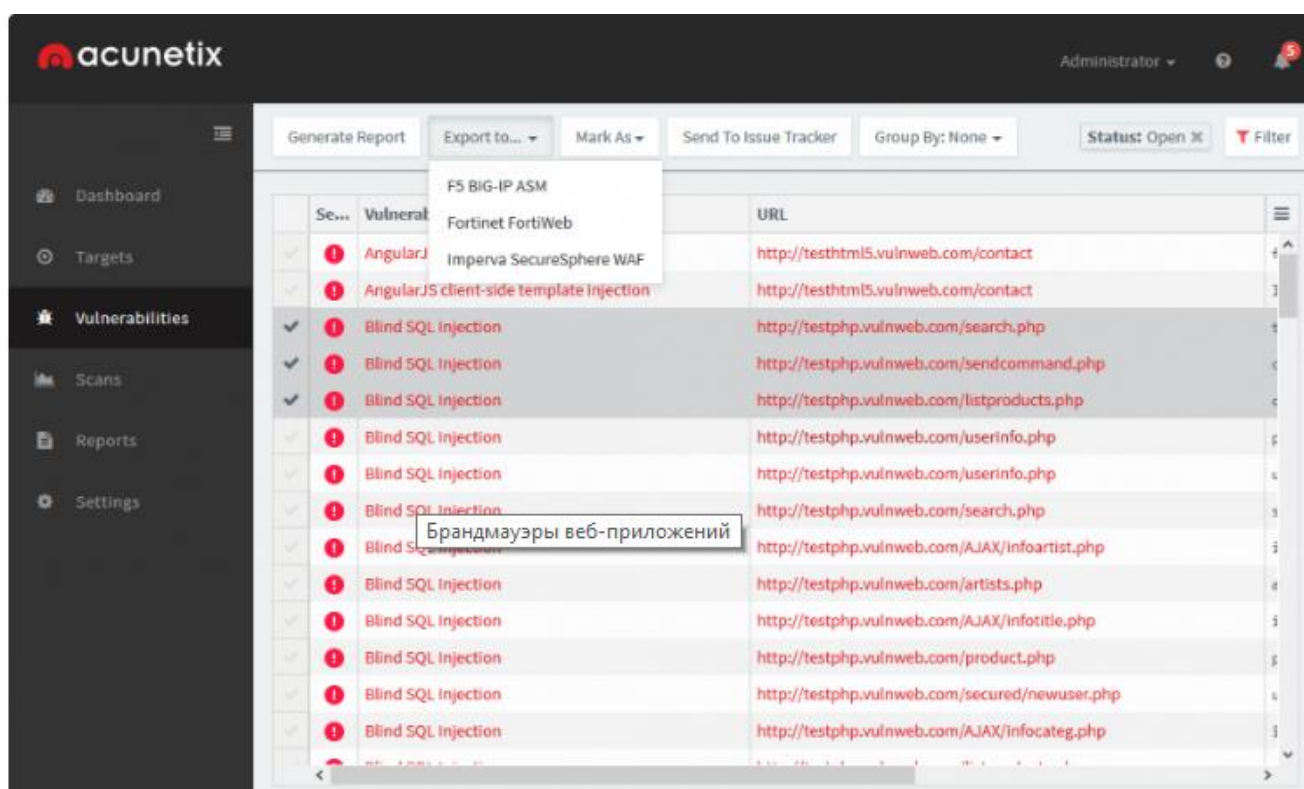


Рисунок 27 – Интерфейс Acunetix Trial

4.6 Автоматизированное тестирование с помощью инструмента Hermione

Автоматизированное тестирование - набор методик, подходов и инструментов, которые позволяют исключить ручное тестирование из выполнения некоторых задач в процессе тестирования. Тестовые случаи частично или полностью выполняются специальным инструментом, однако разработка тестов, подготовка входных данных, оценка результатов выполнения, написание отчетов об ошибках - все это и многое другое выполняет тестировщик.

Автоматизированное тестирование предполагает использование специального программного обеспечения (в дополнение к тестируемому) для мониторинга выполнения тестов для сравнения ожидаемого и фактического результата программы. Этот тип тестирования автоматизирует часто повторяющиеся задачи, которые необходимы для тестирования.

У автоматизированного тестирования есть сильные стороны:

- скорость автоматического тестирования превосходит возможности ручного тестирования;
- отсутствует влияние человеческого фактора;
- минимизация затрат при повторном выполнении тестов;
- способность собирать, хранить, анализировать и представлять большие объемы данных в форме, удобной для человеческого восприятия;
- способность выполнять низкоуровневые действия с приложением, операционной системой, каналами передачи данных и т. д.

Интеграционное тестирование направлено на проверку взаимодействия между несколькими частями приложения (каждая из которых, в свою очередь, проверена отдельно на стадии модульного тестирования). Цель интеграционного тестирования - проверка соответствия проектируемых модулей требованиям функциональности, приема и надежности. Такой уровень тестирования осуществляется через интерфейс [30].

Для автоматизированного интегрального тестирования программного обеспечения WEB-SCADA использовался инструмент Hermione для написания

автоматизированных тестов на JavaScript. Hermione — это утилита для тестирования интеграции веб-страниц на базе WebdriverIO и Mocha, предназначенная для упрощения и ускорения разработки тестов и автоматизации процесса их запуска. Тестирование заключается в том, чтобы воспроизвести действия пользователя в браузере и проверить, корректно ли на них отзывается интерфейс приложения.

Возможности инструмента Hermione:

- простая установка и настройка;
- параллельное выполнение тестов в разных браузерах;
- поддержка пользовательских плагинов;
- гибкая настройка для запуска тестов в отдельных браузерах;
- перезапуск тестов, которые могут дать сбой из-за проблем во внешней инфраструктуре;
- пропуск необходимых тестов в определенных браузерах [31].

Hermione устанавливается как обычный npm-пакет через команду `npm install hermione chai`.

В качестве редактора кода для автоматизированного тестирования использовался Visual Studio Code, разработанный Microsoft для Windows, Linux и macOS. Данный редактор позиционируется как «облегченный» редактор кода для кроссплатформенной разработки веб-приложений.

В редакторе Visual Studio Code создаем проект, в который помещаем файл `.hermione.conf.js` и прописываем следующее содержимое:

```
const testFolder = '2_login'; //название теста
module.exports = {
  gridUrl: 'http://localhost:4444/wd/hub',
  // retry: 2,
  sets: {
    desktop: {
      files: `tests/${testFolder}` //запускаем тесты из папки
    }
  },
  browsers: {
    chrome: {
      sessionsPerBrowser: 1,
      desiredCapabilities: {
        browserName: 'chrome',
```

```

    }
  },

```

В папке проекта необходимо создать сам тестовый сценарий 2_login, который имеет следующий фрагмент кода:

```

const assert = require('assert');
const variables = require('./variables.hermione');
const useVariables = variables._2_login_1_authorization;
const userHermione = require('../user.hermione');

describe('Login | Ввод login и password -> ', () => {

  it ('Поле login - ввод error. Поле password - пустое', function() {
    return this.browser
      .clearSetInput(useVariables.input.login.element, 'error')
  });

  it ('Кнопка Login заблокирована при пустом поле password', function() {
    return this.browser
      .assertGetAttribute(useVariables.buttonLogin.element, 'tabindex', '-1', 'Кнопка Login доступна для нажатия');
  });

  it ('Поле login - пустое. Поле password - error', function() {
    return this.browser
      .clearInput(useVariables.input.login.element)
      .clearSetInput(useVariables.input.password.element, 'error')
  });

  it ('Кнопка Login заблокирована при пустом поле login', function() {
    return this.browser
      .assertGetAttribute(useVariables.buttonLogin.element, 'tabindex', '-1', 'Кнопка Login доступна для нажатия');
  });

  it ('Поле login - error. Поле password - error', function() {
    return this.browser
      .clearSetInput(useVariables.input.login.element, 'error')
      // .click(useVariables.buttonLogin.element)
  });

  it ('Кнопка Login разблокирована при заполнении полей input и password. Неверном заполнении', function() {
    return this.browser

```

```

        .assertGetAttribute(useVariables.buttonLogin.element, '
tabindex', '0', 'Кнопка Login заблокирована для нажатия');
    });

    it ('При неверном заполнении клик по кнопке Login и появлен
ие сообщения об ошибке авторизации', function() {
        return this.browser
            .click(useVariables.buttonLogin.element)
            .waitForExist(useVariables.statusError.element, 5000)
            .assertGetAttribute(useVariables.statusError.element, '
class', useVariables.statusError.class, 'Не появилось сообщение об
ошибке авторизации');
    });

    it ('Поле login - admin. Поле password - 1. Кнопка Login -
> Успешная авторизация', function() {
        return this.browser
            .clearSetInput(useVariables.input.login.element, 'admin
')
            .clearSetInput(useVariables.input.password.element, '1'
)
            .click(useVariables.buttonLogin.element)
            .waitForExist(useVariables.buttonLogin.element, 10000,
true);
    });
});
...

```

После того, как тест написан, его можно запустить. Для этого необходимо сначала запустить сервер Selenium командой `selenium-standalone start`. Сервер будет доступен по адресу `http://localhost:4444` (рисунок 28).

```

PS C:\TESTS\hermione-master\UPD> selenium-standalone start
21:28:59.544 INFO [GridLauncherV3.parse] - Selenium server version: 3.141.5, revision: d54ebd709a
21:28:59.628 INFO [GridLauncherV3.lambda$buildLaunchers$3] - Launching a standalone Selenium Server on port 4444
2020-05-14 21:28:59.686:INFO::main: Logging initialized @431ms to org.seleniumhq.jetty9.util.log.StdErrLog
21:28:59.917 INFO [WebDriverServlet.<init>] - Initialising WebDriverServlet
21:29:00.100 INFO [SeleniumServer.boot] - Selenium Server is up and running on port 4444
Selenium started
_

```

Рисунок 28 – Запуск сервера Selenium

Теперь можно запустить автоматизированное тестирование из папки, где лежит файл `.hermione.config.js`. Для этого выполняется команда `prx hermione` (рисунок 29).

```

PS C:\TESTS\hermione-master\UPD> npx hermione
X Login | Проверка присутствия на странице полей авторизации -> Загрузка окна авторизации [chrome:447fc015e9f750601d7340f401205fa1] - 3736ms
X Login | Проверка присутствия на странице полей авторизации -> Проверка присутствия и доступности для ввода поля admin [chrome:447fc015e9f750601d7340f401205fa1] - 363ms
X Login | Проверка присутствия на странице полей авторизации -> Проверка присутствия и доступности для ввода поля password [chrome:447fc015e9f750601d7340f401205fa1] - 364ms
X Login | Проверка присутствия на странице полей авторизации -> Проверка присутствия и доступности кнопки Login [chrome:447fc015e9f750601d7340f401205fa1] - 343ms
✓ Login | Ввод login и password -> Поле login - ввод error. Поле password - пустое [chrome:447fc015e9f750601d7340f401205fa1] - 4621ms
✓ Login | Ввод login и password -> Кнопка Login заблокирована при пустом поле password [chrome:447fc015e9f750601d7340f401205fa1] - 50ms
✓ Login | Ввод login и password -> Поле login - пустое. Поле password - error [chrome:447fc015e9f750601d7340f401205fa1] - 7652ms
✓ Login | Ввод login и password -> Кнопка Login заблокирована при пустом поле login [chrome:447fc015e9f750601d7340f401205fa1] - 46ms
✓ Login | Ввод login и password -> Поле login - error. Поле password - error [chrome:447fc015e9f750601d7340f401205fa1] - 4540ms
✓ Login | Ввод login и password -> Кнопка Login разблокирована при заполнении полей input и password. Неверном заполнении [chrome:447fc015e9f750601d7340f401205fa1] - 52ms
X Login | Ввод login и password -> При неверном заполнении клик по кнопке Login и появление сообщения об ошибке авторизации [chrome:447fc015e9f750601d7340f401205fa1] - 9638ms
✓ Login | Ввод login и password -> Поле login - admin. Поле password - 1. Кнопка Login -> Успешная авторизация [chrome:447fc015e9f750601d7340f401205fa1] - 9638ms

```

Рисунок 29 – Запуск инструмента Hermione

Далее автоматически открывается браузер, в котором запускается тест и визуально показываются все действия (ввод текста, клики, перемещения и т. д.), тем самым имитируя действия пользователя. На рисунке 30 представлен скриншот процесса тестирования.

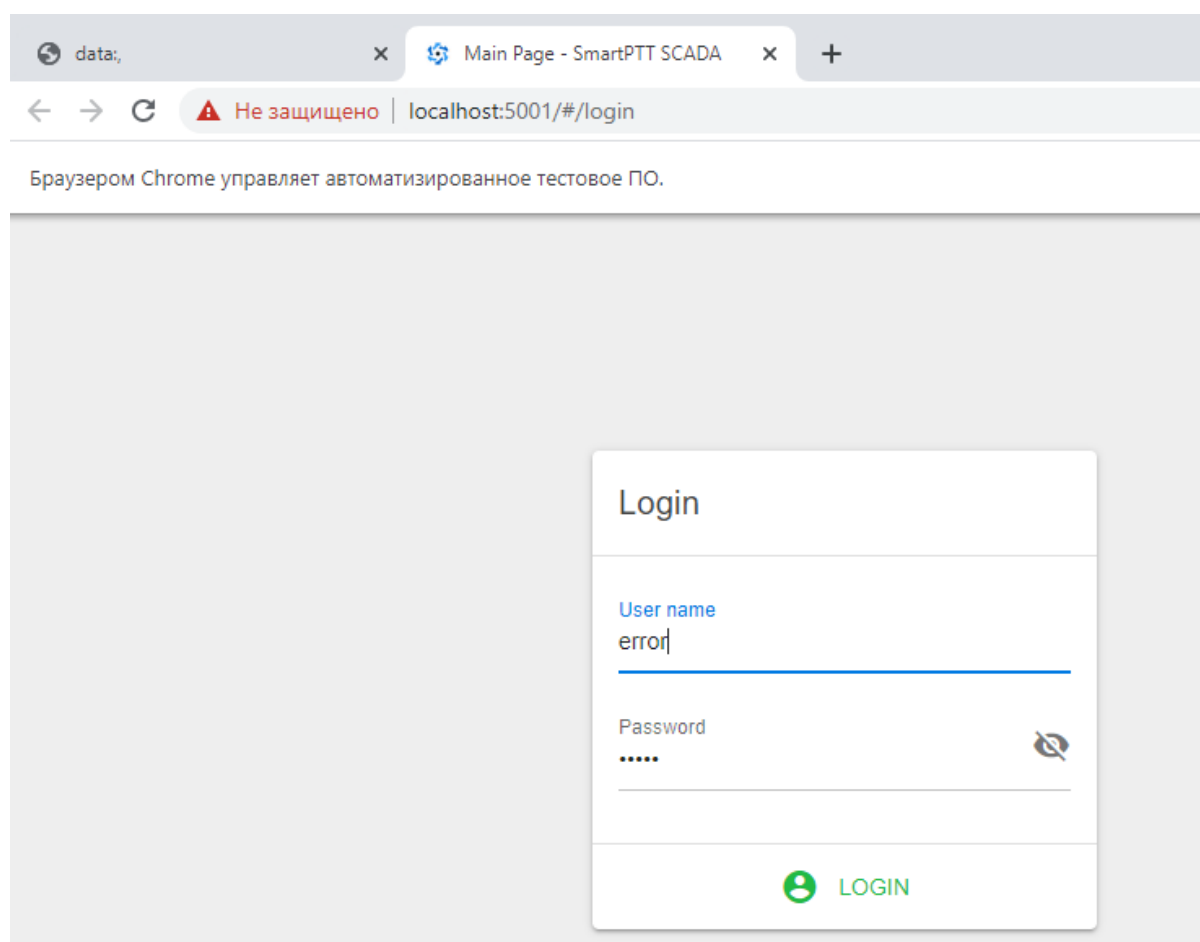


Рисунок 30 – Процесс автоматизированного тестирования

После завершения тестов в папке **hermione-reports** создаются файлы отчёта с помощью плагина **html-reporter**. На рисунке 31 представлена

сформированная папка для отчетов. При каждом запуске тестов отчет обнуляется.

Имя	Дата изменения	Тип
hermione-reports	14.05.2020 21:30	Папка с файлами
images	20.04.2020 12:32	Папка с файлами
node_modules	20.04.2020 11:04	Папка с файлами
tests	27.03.2020 18:18	Папка с файлами
.gitignore	27.03.2020 18:18	Текстовый докум...
.hermione.conf	20.04.2020 12:31	файл JavaScript
package	27.03.2020 18:18	JSON File
package-lock	20.04.2020 11:04	JSON File
README.md	27.03.2020 18:18	Файл "MD"

Рисунок 31 – Запуск сервера Selenium

Затем с помощью браузера запускаем отчет **hermione-reports\index.html**, который показывает подробную информацию (рисунок 32).

Total Tests: 12 Passed: 7 Failed: 5 Skipped: 0 Retries: 0

Show all Expand all Collapse all Expand errors Expand retries Show skipped Show only diff Scale images Lazy load Group by error change original host for view in browser

filter by test name

▼ Login | Ввод login и password ->

- ▶ Кнопка Login заблокирована при пустом поле login
- ▶ Кнопка Login заблокирована при пустом поле password
- ▶ Кнопка Login разблокирована при заполнении полей input и password. Неверном заполнении
- ▼ Поле login - admin. Поле password - 1. Кнопка Login -> Успешная авторизация
- ▶ chrome
- ▶ Поле login - error. Поле password - error
- ▶ Поле login - ввод error. Поле password - пустое
- ▶ Поле login - пустое. Поле password - error
- ▼ При неверном заполнении клик по кнопке Login и появление сообщения об ошибке авторизации
- ▼ chrome

Background: ☒ ☐ ☐

▶ Meta-info

```
message: Cannot read property 'element' of undefined
stack: TypeError: Cannot read property 'element' of undefined
    at Object.<anonymous> (C:\TESTS\hermione-masterUPD\tests\2_login\1_authorization.hermione.js:72:46)
    at Object.tryCatcher (C:\TESTS\hermione-masterUPD\node_modules\bluebird\js\release\util.js:16:23)
    at Object.<anonymous> (C:\TESTS\hermione-masterUPD\node_modules\bluebird\js\release\method.js:15:34)
    at ExecutionThread._call (C:\TESTS\hermione-masterUPD\node_modules\hermione\lib\worker\runner\test-runner\execution-thread.js:46:53)
    at ExecutionThread.run (C:\TESTS\hermione-masterUPD\node_modules\hermione\lib\worker\runner\test-runner\execution-thread.js:28:24)
    at TestRunner.run (C:\TESTS\hermione-masterUPD\node_modules\hermione\lib\worker\runner\test-runner\index.js:48:35)
```

▶ Page screenshot

▼ Login | Проверка присутствия на странице полей авторизации ->

Рисунок 32 – Отчет hermione-reports

Отчет демонстрирует, сколько тестов было запущено, сколько из них прошло успешно, а сколько неудачных. Помимо этого, можно посмотреть причину, почему тест «упал» (неверный ввод логина и пароля, отсутствие полей авторизации и т. д.).

Таким образом, были написаны автоматизированные тесты на JavaScript для всего программного обеспечения WEB-SCADA.

5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

При работе над проектной и научно-исследовательской деятельностью немаловажную роль играет определение экономического обоснования. Раздел «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является обязательной частью магистерской диссертации, целью которого является проектирование и создание конкурентоспособных разработок, технологий, которые отвечают современным требованиям в области ресурсоэффективности и ресурсосбережения.

Задачами данного исследования являются:

- оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения;
- определение возможных альтернатив проведения научных исследований;
- планирование научно-исследовательских работ;
- определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования.

Магистерская диссертация представляет собой разработку и тестирование программного обеспечения для автоматизированного управления процессами с применением WEB-технологий.

5.1 Организация и планирование работ

При организации процесса реализации конкретного проекта необходимо оптимально планировать занятость каждого из его участников и сроки проведения отдельных работ. На данном этапе составляется полный перечень проводимых работ, и определяются их исполнители и оптимальная продолжительность. Результатом планирования работ является сетевой, либо линейный график реализации проекта. Наиболее удобным, простым и наглядным способом для этих целей является использование линейного графика. Для его построения составим перечень работ и соответствие работ своим исполнителям,

продолжительность выполнения этих работ и сведем их в таблицу 4.

Таблица 4 – Перечень работ и продолжительность их выполнения

Этапы работ	Исполнители	Загрузка исполнителей
Подготовительный этап		
Постановка целей и задач, получение исходных данных	НР	НР – 100 %
Составление и утверждение ТЗ	НР, И	НР – 10 % И – 90 %
Разработка календарного плана	НР, И	НР – 100 % И – 100 %
Изучение предметной области	И	И – 100 %
Анализ разработки программного обеспечения		
Разработка и тестирование ПО, приведение полученных результатов	НР, И	НР – 20 % И – 80 %
Экономическое обоснование		
Анализ перспективности разработки, оценка целесообразности разработки и затрат	И	И – 100 %
Анализ опасных и вредных производственных факторов		
Анализ выявленных вредных и опасных факторов производственной среды, защита в чрезвычайных ситуациях, правовые и организационные вопросы обеспечения безопасности	И	И – 100 %
Оформление полученных результатов	И	И – 100 %
Документирование		
Составление и согласование расчетно- пояснительной записки	НР, И	НР – 100 % И – 100 %
Оформление графического материала	И	И – 100 %
Подведение итогов	НР, И	НР – 100 % И – 100 %

5.1.1 Продолжительность этапов работ

Расчет продолжительности этапов работ осуществляется двумя методами:

- технико-экономическим;
- опытно-статическим.

В данном случае используется опытно-статический метод, который реализуется двумя способами:

- аналоговый;
- экспертный.

Для определения ожидаемого значения трудоемкости i -ой работы $t_{ож\ i}$ применяется вероятностный метод – метод двух оценок $t_{min\ i}$ и $t_{max\ i}$:

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5} \quad (5.1)$$

где $t_{min\ i}$ – минимально возможная трудоемкость выполнения заданной i -ой работы, чел/дн.;

$t_{max\ i}$ – максимально возможная трудоемкость выполнения заданной i -ой работы, чел/дн.

Для выполнения перечисленных в таблице 4 работ требуются специалисты:

- инженер-разработчик ПО;
- научный руководитель.

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести в календарные дни.

Расчет продолжительности выполнения каждого этапа в рабочих днях ведется по формуле:

$$T_{р\ д} = t_{ож\ i} \cdot K_{д}, \quad (5.2)$$

где $t_{ож}$ – продолжительность работы, дн.;

$K_{д}$ – коэффициент, учитывающий дополнительное время на компенсации и согласование работ ($K_{д} = 1.2$).

Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{\text{КД}} = T_{\text{РД}} \cdot T_{\text{К}}, \quad (5.3)$$

где $T_{\text{РД}}$ – продолжительность выполнения этапа в рабочих днях;

$T_{\text{КД}}$ – продолжительность выполнения этапа в календарных днях; $T_{\text{К}}$ – коэффициент календарности.

Коэффициент календарности рассчитывается по формуле:

$$T_{\text{К}} = \frac{T_{\text{КАЛ}}}{T_{\text{КАЛ}} - T_{\text{ВД}} - T_{\text{ПД}}} \quad (5.4)$$

где $T_{\text{КАЛ}}$ – календарные дни ($T_{\text{КАЛ}} = 365$); $T_{\text{ВД}}$ – выходные дни ($T_{\text{ВД}} = 52$); $T_{\text{ПД}}$ – праздничные дни ($T_{\text{ПД}} = 10$).

В таблице 5 приведены длительность этапов работ и число исполнителей, занятых на каждом этапе:

$$T_{\text{К}} = \frac{365}{365 - 52 - 10} = 1,205$$

Таблица 5 – Трудозатраты на выполнение проекта

Этапы работ	Исполнители	Продолжительность работ, дни			Длительность работ, чел/дн			
					ТРД		ТКД	
		<i>tmin i</i>	<i>tmax i</i>	<i>тож i</i>	НР	И	НР	И
Подготовительный этап								
Постановка целей и задач, получение исходных данных	НР	3	5	3,8	4,56	-	5,49	-
Составление и утверждение требований к ПО	НР, И	4	6	4,8	0,58	5,18	0,69	6,24
Разработка календарного плана	НР, И	1	2	1,4	1,68	1,68	2,02	2,02
Изучение предметной области	И	7	10	8,2	-	9,84	-	11,86
Анализ разработки программного обеспечения								
Разработка и тестирование ПО, приведение полученных результатов	НР, И	14	154	70,0	16,80	84,00	20,24	101,22
Оформление полученных результатов	И	5	7	5,8	-	6,96	-	8,39
Экономическое обоснование								
Анализ перспективности разработки, оценка целесообразности разработки и затрат	И	3	5	3,8	-	4,56	-	5,49
Анализ опасных и вредных производственных факторов								
Анализ выявленных вредных и опасных факторов производственной среды, защита в чрезвычайных ситуациях, правовые и организационные вопросы обеспечения безопасности	И	1	2	1,4	-	1,68	-	2,02
Документирование								
Составление и согласование расчетно-пояснительной записки	НР, И	3	5	3,8	4,56	4,56	5,49	5,49
Оформление графического материала	И	3	5	3,8	-	4,56	-	5,49
Подведение итогов	НР, И	1	2	1,4	1,68	1,68	2,02	2,02
Всего:				108,2	29,86	124,70	35,95	150,24

На основе таблицы 5 строится календарный план-график. График строится для максимального по длительности исполнения работ в рамках научно-исследовательского проекта на основе таблицы 2 с разбивкой по месяцам и декадам за период времени дипломирования. При этом работы на графике следует выделить различной штриховкой в зависимости от исполнителей, ответственных за ту или иную работу. На рисунке 33 представлен линейный график работ.

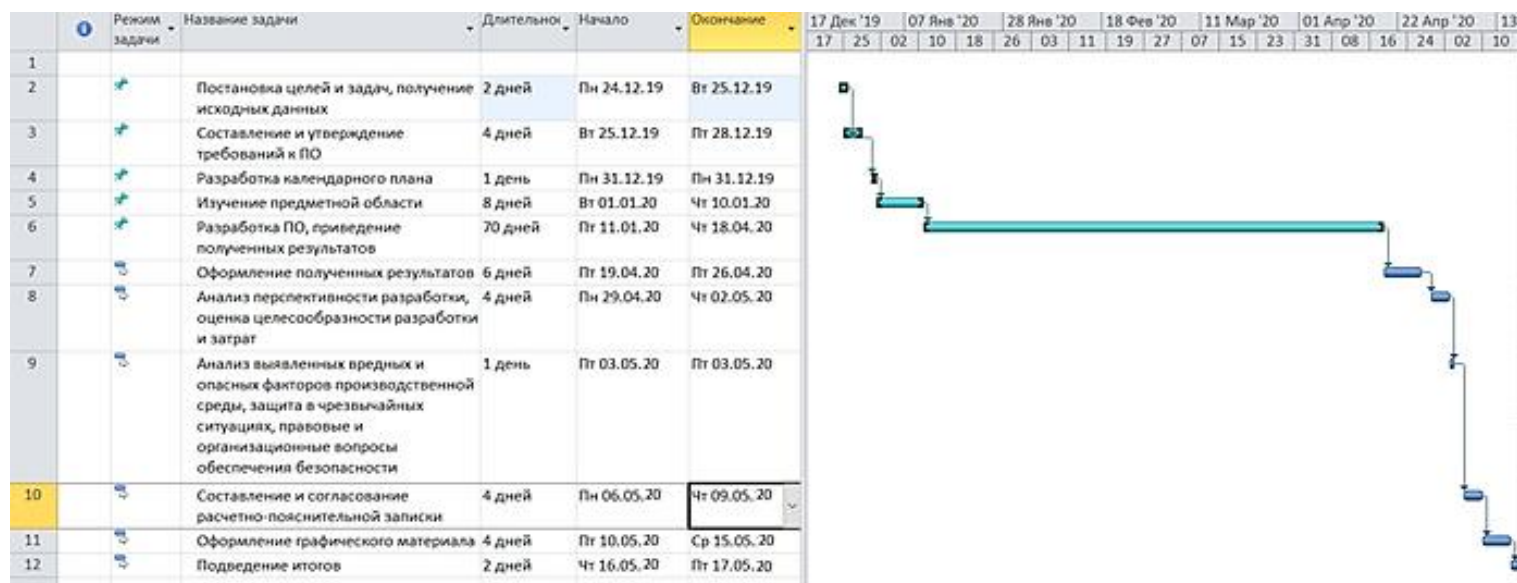


Рисунок 33 – Линейный график работ

5.2 Расчет сметы затрат на выполнение проекта

В состав затрат на создание проекта включается стоимость всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчет сметной стоимости на выполнение данной разработки производится по следующим статьям затрат:

- материалы и покупные изделия;
- заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- прочие (накладные расходы) расходы.

5.2.1 Расчет затрат на материалы

К данной статье расходов относится стоимость материалов, покупных изделий, полуфабрикатов и других материальных ценностей, расходуемых непосредственно в процессе выполнения работ. Цена материальных ресурсов определяется по соответствующим ценникам и приведена в таблице 6.

Таблица 6 – Расчет затрат на материалы

Наименование материалов	Цена за ед., руб.	Количество	Сумма, руб.
Бумага для принтера формата А4	190	1 уп.	190
Картридж для принтера	1 550	1 шт.	1550
Итого:			1740

Допустим, что ТЗР составляют 5 % от отпускной цены материалов, тогда расходы на материалы с учетом ТЗР равны $C_{\text{мат}} = 1740 * 1,05 = 1827 \text{руб.}$

5.2.2 Расчет заработной платы

Данная статья расходов включает заработную плату научного руководителя и инженера, а также премии, входящие в фонд заработной платы. Расчет основной заработной платы выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя.

Среднедневная тарифная заработная плата ($ЗП_{\text{дн-т}}$) рассчитывается по формуле:

$$ЗП_{\text{дн-т}} = MO / 25,083 \quad (5.5)$$

Расчеты затрат на основную заработную плату приведены в таблице 7. При расчете учитывалось, что в году 298 рабочих дней и, следовательно, в месяце 25,083 рабочих дня. Затраты времени на выполнение работы по каждому исполнителю брались из таблицы 5. Также был принят во внимание учитывающий коэффициент дополнительной заработной платы $КПР = 1,1$; $К_{\text{доп.ЗП}} = 1,188$; $К_r = 1,3$. Таким образом, для перехода от тарифной (базовой) суммы заработка исполнителя, связанной с участием в проекте, к соответствующему полному заработку (зарплатной части сметы) необходимо

первую умножить на интегральный коэффициент $K_{\text{и}} = 1,1 \cdot 1,188 \cdot 1,3 = 1,699$.
Считаем, что И и НР работают в неделю 6-ть рабочих дней.

Таблица 7 - Затраты на заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./день	Затраты времени, дни	Коэффици- ент	Фонд з/платы, руб.
НР	23 264,86	927,51	30	1,699	47275,18
И	14 584,32	581,44	125	1,699	123483,32
Итого:					170758,50

Таким образом, затраты на основную заработную плату составили:

$$C_{\text{ОСН}} = 170758,50 \text{ руб.}$$

5.2.3 Расчет затрат на социальный налог

Затраты на единый социальный налог (ЕСН), включающий в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, составляют 30 % от полной заработной платы по проекту, т.е.

$$C_{\text{СОЦ}} = K_{\text{СОЦ}} \cdot C_{\text{ОСН}} \quad C_{\text{соц.}} = C_{\text{зп}} \cdot 0,3 \quad (5.6)$$

Таким образом, в нашем случае $C_{\text{соц.}} = 170758,50 \cdot 0,3 = 51227,55 \text{ руб.}$

5.2.4 Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, которая была потрачена в ходе выполнения проекта на работу используемого оборудования. Рассчитываются по формуле:

$$C_{\text{эл.об.}} = P_{\text{об}} \cdot t_{\text{об}} \cdot \text{Ц}_{\text{э}} \quad (5.7)$$

где $P_{\text{ОБ}}$ – мощность, потребляемая оборудованием, кВт;

$\text{Ц}_{\text{э}}$ – тариф на 1 кВт·час;

$t_{\text{об}}$ – время работы оборудования, час.

Для ТПУ $\text{Ц}_{\text{э}} = 6,59 \text{ руб./кВт·час}$ (с НДС).

Время работы оборудования вычисляется на основе итоговых данных таблицы 5.2 для инженера ($T_{\text{рд}}$) из расчета, что продолжительность рабочего дня

равна 8 часов.

$$t_{об} = T_{рД} * K_t, \quad (5.8)$$

где $K_t \leq 1$ – коэффициент использования оборудования по времени.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{об} = P_{ном.} * K_C \quad (5.9)$$

где $P_{ном.}$ – номинальная мощность оборудования, кВт;

$K_C \leq 1$ – коэффициент загрузки, зависящий от средней степени использования номинальной мощности.

Расчет затраты на электроэнергию для технологических целей приведен в таблице 8.

Таблица 8 – Затраты на электроэнергию

Наименование оборудования	Время работы оборудования $t_{об}$, час	Потребляемая мощность $P_{об}$, кВт	Затраты Эоб, руб.
Персональный компьютер	997,6*0,6	0,3	1183,35
Струйный принтер	30	0,1	17,24
Итого:			1200,59

5.2.5 Расчет амортизационных расходов

В статье «Амортизационные отчисления» от используемого оборудования рассчитывается амортизация за время выполнения работы для оборудования, которое имеется в наличии. Амортизационные отчисления рассчитываются на время использования ЭВМ по формуле (10):

$$C_{ам} = \frac{N_A * C_{об} * t_{рф} * n}{F_d}, \quad (5.10)$$

где N_A – годовая норма амортизации;

$C_{об}$ – цена оборудования (персональный компьютер);

F_d – действительный годовой фонд рабочего времени;

$t_{рф}$ – фактическое время работы оборудования при создании программного продукта;

n – число задействованных ПЭВМ.

При использовании нескольких типов оборудования расчет по формуле делается соответствующее число раз, затем результаты суммируются.

Стоимость ПК 64500 руб, время использования 997,6 часа, тогда для него $C_{AM}(ПК) = (0,4 \cdot 64500 \cdot 997,6 \cdot 1) / 2408 = 10688,57$ руб. Стоимость принтера 12000 руб., его $F_d = 500$ час.; $N_A = 0,5$; $t_{pf} = 30$ час., тогда его $C_{AM}(Пр) = (0,5 \cdot 12000 \cdot 30 \cdot 1) / 500 = 360$ руб.

Итак, затраты на амортизационные отчисления составили 11048,57 руб.

5.2.6 Расчет прочих расходов

В статье «Прочие расходы» отражены расходы на выполнение проекта, которые не учтены в предыдущих статьях, их следует принять равными 10% от суммы всех предыдущих расходов, т.е.

$$C_{\text{проч.}} = (C_{\text{мат}} + C_{\text{зп}} + C_{\text{соц}} + C_{\text{эл.об.}} + C_{\text{ам}}) \cdot 0,1 \quad (5.11)$$

Для данного примера получаем:

$$C_{\text{проч.}} = (1827 + 170758,5 + 51227,55 + 1200,59 + 11048,57) \cdot 0,1 = 23597,52 \text{ руб.}$$

5.2.7 Расчет общей себестоимости разработки

Проведя расчет по всем статьям сметы затрат на разработку, можно определить общую себестоимость проекта «Разработка и тестирование системы автоматизированного управления» (таблица 9).

Таблица 9 – Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Материалы и покупные изделия	$C_{\text{мат}}$	1827
Основная заработная плата	$C_{\text{зп}}$	170758,5
Отчисления в социальные фонды	$C_{\text{соц}}$	51227,55
Расходы на электроэнергию	$C_{\text{эл.}}$	1200,59
Амортизационные отчисления	$C_{\text{ам}}$	11048,57
Прочие расходы	$C_{\text{проч}}$	23597,52
Итого:		259572,73

Таким образом, затраты на разработку составили $C = 259572,73$ руб.

5.2.8 Расчет прибыли

Прибыль от реализации проекта в зависимости от конкретной ситуации может определяться различными способами. Если исполнитель работы не располагает данными для применения «сложных» методов, то прибыль следует принять в размере $5 \div 20 \%$ от полной себестоимости проекта. В данном примере она составляет 51914,55 руб. (20 %) от расходов на разработку проекта.

5.2.9 Расчет НДС

НДС составляет 20% от суммы затрат на разработку и прибыли. В нашем случае это $(259572,73 + 51914,55) * 0,2 = 62297,45$ руб.

5.2.10 Цена разработки НИР

Цена равна сумме полной себестоимости, прибыли и НДС. В данном случае

$$C_{\text{НИР(КР)}} = 259572,73 + 51914,55 + 62297,45 = 373784,74 \text{ руб.}$$

5.3 Оценка экономической эффективности проекта

Важным аспектом качества завершенного проекта является экономическая эффективность его реализации, т. е. соотношение обусловленного им экономического результата (эффекта) и затрат на разработку проекта. Поскольку последние являются единовременными, то мы имеем дело с частным случаем задачи оценки экономической эффективности инвестиций, то есть инвестирования денег в предприятие, организацию, отрасль, региональную социально-экономическую систему и т. д. (так называемые инвестиционные объекты) с целью получения определенного результата в будущем.

Данную работу можно измерить по прямым экономическим показателям, таких как: сокращение рабочего времени при эксплуатации объекта и снижения рисков нарушения нормального режима работы.

В нашем случае прямым показателем будет уменьшение времени работы

инженера при настройке автоматизированного тестирования WEB-Scada с помощью инструмента Hermione. При использовании инструмента интегрального тестирования Hermione временные затраты могут увеличиваться в зависимости от изменения интерфейса приложения WEB-Scada в течении года, или в более короткий срок. Другими словами, инструмент Hermione становится непригодным для тестирования приложения в данный момент времени, что заставляет возвращаться к переписыванию кода для интегрального тестирования снова. При изменении интерфейса селекторы DOM-дерева меняются, и инструмент Hermione не может их найти.

Количественная оценка эффективности невозможна ввиду недостатка информации и компетенции, однако, данная методика, которая заключается в присвоении id конкретным form-полям, позволяет актуализировать переменные DOM-дерева для Hermione и в будущем не тратить время на актуализацию тестов. Таким образом, данный подход избавит инженера от необходимости постоянной актуализации интегрального тестирования, что, в свою очередь, уменьшает суммарное время обслуживания тестов и время работы. За счет сокращения временных затрат на проектирование и реализацию также уменьшатся расходы на электроэнергию.

Помимо этого, при использовании методов и видов тестирования, примененных в работе, можно исключить риск возникновения ошибок в режиме работы ПО. Положительный результат использованных методов: соответствие между ожидаемым поведением программы и её реальным поведением. Если же реальное поведение программы не соответствует заявленным результатам, то заказчики возвращают низкокачественный продукт, а предприятие терпит издержки. Таким образом, данный подход снижает риски нарушения работы из-за некачественного тестирования, чем помогает в экономии ресурсов предприятия.

6. Социальная ответственность

Объектом исследования является программное обеспечение, предназначенное для разработки или поддержания работы в реальном времени систем сбора, обработки и отображения информации об объекте управления и мониторинга.

В связи с этим, в работе рассматривается направления и процесс тестирования, проводится анализ разрабатываемого приложения, а также разработка автоматизированного тестирования с помощью инструмента Hermione.

В данном разделе будет разработан комплекс мероприятий, который позволит свести к минимуму или ликвидировать негативные влияния факторов, возникающих при разработке и тестировании программного обеспечения на ПЭВМ.

Магистерская диссертация выполнялась в десятом корпусе ТПУ (первый этаж) в отделении информационных технологий.

6.1 Правовые и организационные вопросы обеспечения безопасности

6.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства

При организации рабочего места основной целью является обеспечение качественного и эффективного выполнения работы при полном использовании оборудования в соответствии с установленными сроками [32]. В связи с этим требования к рабочему месту носят следующий характер:

- 1) Рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 [33], ГОСТ 12.2.061-81 [34] и СанПиН 2.2.2/2.4.1340-03 [35];

- 2) Конструкция рабочей мебели (рабочий стол, кресло, подставка для ног) должна обеспечивать возможность индивидуальной регулировки соответственно росту пользователя и создавать удобную позу для работы.

Вокруг ЭВМ должно быть обеспечено свободное пространство не менее 60-120см;

3) На уровне экрана должен быть установлен оригинал-держатель.

На рисунке 34 представлены требования к рабочему месту.



Рисунок 34 – Организация рабочего места

6.1.2 Организационные мероприятия при компоновке рабочей зоны

Рабочее место должно быть организовано в соответствии с требованиями стандартов, технических условий и (или) методических указаний по безопасности труда. Оно должно удовлетворять следующим требованиям:

- обеспечивать возможность удобного выполнения работ;
- учитывать физическую тяжесть работ;
- учитывать размеры рабочей зоны и необходимость передвижения в ней работающего;
- учитывать технологические особенности процесса выполнения работ.

Невыполнение требований к расположению и компоновке рабочего места может привести к получению работником производственной травмы или развития у него профессионального заболевания.

В соответствии с государственными стандартами и правовыми нормами

обеспечения безопасности предусмотрена рациональная организация труда в течение смены, которая предусматривает:

- длительность рабочей смены не более 8 часов;
- установление двух регламентируемых перерывов (не менее 20 минут после 1-2 часов работы, не менее 30 минут после 2 часов работы);
- обеденный перерыв не менее 40 минут.

Кроме того, предусмотрен предварительный медосмотр при приеме на работу и периодические медосмотры.

Также перед приемом на работу каждый сотрудник должен пройти инструктаж по технике безопасности, по электробезопасности и охране труда.

6.2 Профессиональная социальная ответственность

6.2.1 Анализ вредных и опасных факторов, которые может создать объект исследования

На оператора ПЭВМ в течение рабочего дня воздействует множество различных производственных факторов, каждый из которых влияет на производительность, работоспособность и физическое состояние.

При разработке устройства могут возникнуть вредные и опасные факторы. Используя ГОСТ 12.0.003-74 [36], можно выделить ряд факторов, приведенных в таблице 10. Также приведены источники факторов и нормативные документы, регламентирующие действие каждого фактора.

Таблица 10 – Опасные и вредные факторы при разработке устройства

Источник фактора	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
Персональный компьютер	<ul style="list-style-type: none"> • нарушение параметров микроклимата; • недостаточное освещение; • повышение шума; • электромагнитные излучения 	опасность поражения электрическим током	1. СанПиН 2.2.4.548–96 2. СН 2.2.4/2.1.8.562–96 3. СанПиН 2.2.2/2.4.1340-03 4. СанПиН 2.2.4.1191-03 5. ГОСТ Р 12.1.019-2009 ССБТ

6.2.2 Обоснование мероприятий по защите персонала предприятия от действия опасных и вредных факторов

Микроклимат рабочего помещения

Микроклимат производственных помещений – климат внутренней среды этих помещений, который определяется действующими на организм человека сочетаниями температуры, влажности и скорости движения воздуха.

Человек постоянно находится в процессе теплового взаимодействия с окружающей его рабочее место средой. Температура, относительная влажность и скорость движения окружающего воздуха характеризуют процесс теплообмена. Данные параметры оказывают комплексное воздействие на процесс теплообмена на рабочем месте.

В соответствии с СанПиН 2.2.2/2.4.1340-03 [37] в производственных помещениях, в которых работа с использованием ПЭВМ является основной и связана с нервно-эмоциональным напряжением, должны обеспечиваться оптимальные параметры микроклимата в соответствии с действующими санитарно–эпидемиологическими нормативами микроклимата производственных помещений.

В соответствии с СанПиН 2.2.4.548–96 [38], показателями, характеризующими микроклимат в производственных помещениях, являются:

- температура воздуха;
- температура поверхностей;
- относительная влажность воздуха;
- скорость движения воздуха;
- интенсивность теплового облучения.

В таблицах 11 и 12 приведены оптимальные и допустимые величины показателей микроклимата на рабочих местах производственных помещений для оператора ЭВМ. Работа в данном случае относится к категории лёгких работ.

Таблица 11 – Оптимальные величины показателей микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, С ⁰	Температура поверхностей, С ⁰	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	21-23	20-24	40-60	0,1
Теплый	23-25	22-26	40-60	0,1

Оптимальные микроклиматические условия обеспечивают общее и локальное ощущение теплового комфорта при минимальном напряжении механизмов терморегуляции, не вызывают отклонений в состоянии здоровья, создают предпосылки для высокого уровня работоспособности и являются предпочтительными на рабочих местах.

Исходя из требований, рассматриваемых в данном разделе нормативных документов, в использованном помещении поддерживается температура равная 19–20 С°, при относительной влажности в 55–58%. Для этого в помещении проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы с ПЭВМ. Помимо этого, в теплое время года в помещении функционирует система принудительной вентиляции. В зимнее время в помещении предусмотрена система водяного отопления со встроенными нагревательными элементами и терморегуляторами.

Таблица 12 – Допустимые величины показателей микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин			Температура воздуха ниже оптимальных величин	Температура воздуха выше оптимальных величин
Холодный	19,0- 20,9	23,1 - 24,0	18,0 - 25,0	15 - 75	0,1	0,2
Теплый	20,0- 21,9	24,1 - 28,0	19,0 - 29,0	15 - 75	0,1	0,3

Производственное освещение

Естественное и искусственное освещение рабочего места оказывает влияние на физическое состояние и на работу сотрудника. Недостаточный уровень освещенности в помещении приводит к снижению остроты зрения, головным болям, снижению концентрации внимания и, как следствие, к ухудшению производительности труда.

Причиной недостаточной освещенности являются недостаточность естественного освещения, недостаточность искусственного освещения, пониженная контрастность.

Рабочее помещение имеет как естественное, так и искусственное освещение. Коэффициент естественного освещения не менее 1,2%. Освещенность на поверхности рабочего стола в зоне размещения документа 300 – 500 лк (согласно СанПиН 2.2.2/2.4.1340-03 [39]). При освещении на поверхности экрана блики должны отсутствовать. Поверхность экрана должна быть до 300 лк.

Искусственное освещение в помещениях для эксплуатации ПЭВМ должно осуществляться системой общего равномерного освещения. Помимо этого, рабочие места следует размещать так, чтобы естественный свет падал преимущественно слева, а дисплеи монитора были ориентированы боковой стороной к световым проемам.

Расчет искусственного освещения

Задачей расчета является выполнение общего равномерного освещения помещения. Размеры используемого помещения 7х6х4 м, высота рабочей поверхности $h_{rp} = 0,8$ м. Требуется создать освещенность $E = 300$ лк.

Световой поток лампы определяется по формуле:

$$\Phi = \frac{E_m \cdot S \cdot Z \cdot K_z}{N \cdot \eta} \quad (6.1)$$

Коэффициент отражения стен $R_c = 50 \%$, потолка $R_n = 70 \%$. Коэффициент запаса $k = 1,8$, коэффициент неравномерности $Z = 1,1$. Рассчитываем систему

общего люминесцентного освещения. Выбираем светильники типа ОД, $\lambda = 1,4$.
Приняв $h_c = 0,5$ м, получаем:

$$h = 4 - 0,5 - 0,8 = 2,7 \text{ м};$$

$$L = 1,4 * 2,7 = 3,78 \text{ м};$$

$$L/3 = 1,26 \text{ м}.$$

В аудитории № 117 светильники типа ОД установлены в три ряда (по 3 светильника на каждый ряд мощностью 40 Вт). Учитывая, что в каждом светильнике установлено две лампы, общее число люминесцентных ламп ЛТБ в помещении $N = 18$.

Находим индекс помещения: $i = 120 / [2,7 * (15 + 8)] = 1,2$.

Коэффициент использования светового потока: $\eta = 0,61$.

$$\Phi = \frac{300 \cdot 42 \cdot 1,8 \cdot 1,1}{18 \cdot 0,61} = 2299,45 \text{ Лм.} \quad (6.2)$$

Определяем потребный световой поток ламп в каждом из рядов, ближайшая стандартная лампа – ЛТБ 40 Вт с потоком 2580 лм. Делаем проверку выполнения условия:

$$-10 \% \leq (\Phi_{ст} - \Phi_{расч}) / \Phi_{ст} * 100 \% \leq 20 \% \quad (6.3)$$

$$-10 \% \leq (2580 - 2299,45) / 2580 * 100 \% \leq 20 \%$$

$$-10 \% \leq 10,9 \% \leq 20 \%$$

Определяем электрическую мощность осветительной установки
 $P = 18 * 40 = 720 \text{ Вт}.$

В результате анализа освещенности рабочего места отклонений от норм выявлено не было. Уровень освещенности соответствует нормам в разные периоды светового дня. Использованное аудиторного помещения десятого корпуса ТПУ (117 аудитория) полностью соответствует требованиям СанПиН 2.2.2/2.4.1340-03 [39] к организации освещения рабочего места.

Производственные шумы

Шум – это совокупность звуков, неблагоприятно воздействующих на организм человека и мешающих его работе и отдыху. Источниками звука

являются колебания материальных частиц и тел, передаваемых жидкой, твердой и газообразной средой.

При работе с ЭВМ возникают акустические поля. Воздействие шума может привести к ухудшению слуха. Шумовое загрязнение среды на рабочем месте приводит к снижению внимания персонала, замедлению скорости психических реакций. Шумовой фон в помещении возникает из-за работы десяти компьютеров, а также из-за принтеров, телефонов и систем вентиляции. Длительное воздействие этих шумов отрицательно сказывается на эмоциональном состоянии персонала.

Для избегания вышеуказанных последствий воздействия описываемого фактора, необходимо соблюдать следующие требования, обозначенные в СанПиН 2.2.2/2.4.1340–03 [39]. В таблице 13 приведены допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ.

Таблица 13 – Допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ

Уровни звукового давления в октавных полосах со среднегеометрическими частотами (Гц)								
31,5	63	125	250	500	1000	2000	4000	8000
Уровни звука в дБА								
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ

Уровень звука и звукового давления измеряется на расстоянии 50 см от поверхности оборудования и на высоте источника (-ов) звука. На рабочем месте уровень шума является допустимым, так как он не вызывает серьезного беспокойства и не влияет на психологическое состояние.

Электромагнитные поля

Источниками повышенной напряженности электромагнитного поля в данном случае является персональный компьютер. Объясняется это тем, что ПК

оснащают сетевыми фильтрами, источниками бесперебойного питания и другим оборудованием, что в совокупности формирует сложную электромагнитную обстановку на рабочем месте пользователя.

Согласно СанПиН 2.2.4.1191–03 [40], 8-ми часовой рабочий день для сотрудника на своем рабочем месте, с предельно допустимым уровнем напряженности должен составлять не более 8 кА/м, а уровень магнитной индукции – 10 мТл. Соблюдение данных норм дает возможность избежать негативного воздействия электромагнитных излучений.

Для уменьшения уровня электромагнитного поля от персонального компьютера рекомендуется включать в одну розетку не более двух компьютеров, сделать защитное заземление, подключать компьютер к розетке через нейтрализатор электрического поля.

Длительное воздействие электромагнитного поля на организм человека может привести к нарушению дыхательной, нервной и сердечно-сосудистой систем, в дальнейшем появлению головных болей и усталости. Для обеспечения более низкого уровня электромагнитного излучения на рабочем месте используется жидкокристаллический монитор. Помимо этого, компьютер заземлен. Также в период работы с компьютером по возможности предусматривается сокращение времени, затрачиваемого на разработку и тестирование.

Электробезопасность

Электробезопасность - это целый комплекс мер, нормативных документов, нормативных актов и средств защиты, который направлен на снижение вероятности воздействия на людей электрического тока, статического электричества, электромагнитного поля и других факторов, которые могут стать причиной травм различной степени тяжести и даже привести к смерти.

Опасные и вредные воздействия на людей электрического тока, электрической дуги и электромагнитных полей проявляются в виде

электрических травм и профессиональных заболеваний.

Поражение током может произойти в следующих случаях:

- при прикосновении к токоведущим частям во время ремонта ПЭВМ;
- при однофазном (униполярным) касанием незаизолированного человека от земли к незаизолированным токоведущим частям электрических установок, находящихся под напряжением;
- при прикосновении к токоведущим частям, находящимся под напряжением, то есть в случае повреждения изоляции;
- при контакте с полом и стенами, которые оказались под напряжением;
- в случае возможного короткого замыкания в высоковольтных блоках: блок питания, блок развертки монитора.

Помещение кабинета по электробезопасности сухое, хорошо отапливаемое помещение с токонепроводящими полами, с температурой 18-21° и влажностью 40-50, поэтому оно относится к помещению без повышенной опасности, согласно ГОСТ Р 12.1.019-2009 ССБТ [41].

Нормы на допустимые токи и напряжения прикосновения в электроустановках должны устанавливаться в соответствии с предельно допустимыми уровнями воздействия на человека токов и напряжений прикосновения и утверждаться в установленном порядке.

Электробезопасность обеспечивается конструкцией электроустановок; техническими способами и средствами защиты; организационными и техническими мероприятиями.

Для обеспечения защиты от случайного прикосновения к токоведущим частям необходимо применять следующее:

- изоляция токопроводящих частей;
- защитное заземление;
- зануление;
- защитное отключение;

- предупредительная сигнализация и блокировки.

На рабочем месте администратора запрещается прикасаться к тыльной стороне дисплея, вытирать пыль с компьютера при его включенном состоянии, работать на компьютере во влажной одежде и влажными руками.

Помимо этого, проводится ряд организационных мероприятий (специальное обучение, аттестация и переаттестация лиц электротехнического персонала, инструктажи и т. д.).

6.3 Экологическая безопасность

Так как объект исследования данной работы разрабатывался на ПЭВМ, то сама разработка не влечет за собой негативных воздействий на окружающую среду, поэтому создание санитарно-защитной зоны и принятие мер по защите атмосферы, гидросферы, литосферы не являются необходимыми.

Единственными исключениями являются использование персонального компьютера в качестве твердых отходов и, как следствие, загрязнение почвы или выбросы в атмосферу загрязняющих веществ, углекислого газа и выделение тепла в случае пожара.

В конце срока службы ПК его можно отнести к отходам электронной промышленности. Переработка таких отходов осуществляется путем разделения на однородные компоненты, пригодных для дальнейшего использования, и их передачи для дальнейшего использования (например, кремния, алюминия, золота, серебра, редких металлов).

В соответствии с требованиями закона все отходы, образованные в соответствии с их классами опасности, передаются на специализированные предприятия для переработки, утилизации или захоронения.

Отходы, которые не подлежат переработке или повторному использованию, должны быть утилизированы на полигонах или в почве.

6.4 Безопасность в чрезвычайных ситуациях

6.4.1 Анализ вероятных ЧС, которые могут возникнуть на производстве при разработке и эксплуатации объекта исследований

Чрезвычайная ситуация — это состояние, при котором в результате возникновения источника ЧС на объекте, определенной территории или акватории нарушаются нормальные условия жизни и деятельности людей, возникает угроза их жизни и здоровью, наносится ущерб имуществу населения, народному хозяйству и природной среде.

Наиболее характерной ЧС для помещения, оборудованных ЭВМ, является пожар.

Причинами возникновения данного вида ЧС являются:

- возникновением короткого замыкания в электропроводке;
- возгоранием устройств ПК из-за неисправности аппаратуры;
- возгоранием устройств искусственного освещения;
- возгоранием мебели по причине нарушения правил пожарной безопасности, а также неправильного использования дополнительных бытовых электроприборов и электроустановок.

6.4.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС

Пожарная безопасность - совокупность организационно-технических мероприятий, направленных на обеспечение безопасности людей, предотвращение пожара, ограничение его распространения, а также создание условий для успешного тушения пожаров.

Наиболее типичной чрезвычайной ситуацией для нашего объекта является пожар. Эта аварийная ситуация может возникнуть в случае короткого замыкания в проводке оборудования, обрыва провода, несоблюдения мер пожарной безопасности в офисе и т. д.

Следующие меры относятся к противопожарным мерам в помещении:

1. помещение должно быть оборудовано: средствами тушения пожара (огнетушителями, ящиком с песком, стендом с противопожарным инвентарем); средствами связи; должна быть исправна электрическая проводка осветительных приборов и электрооборудования.

2. каждый сотрудник должен знать место нахождения средств пожаротушения и средств связи; помнить номера телефонов для сообщения о пожаре и уметь пользоваться средствами пожаротушения.

Помещение обеспечено средствами пожаротушения в соответствии с нормами и иметь: пенный огнетушитель ОП-10 – 1 шт; углекислотный огнетушитель ОУ-5 – 1 шт.

Принудительная эвакуация при пожаре происходит в условиях усиливающегося действия опасных факторов пожара. Короткая продолжительность процесса аварийной эвакуации достигается наличием аварийных маршрутов и выходов, количество, размеры и конструктивно-планировочные решения которых регламентированы строительными нормами СНиП 2.01.02-85 [42].

Для предотвращения возникновения пожара необходимо проводить следующие профилактические работы, направленные на устранение возможных источников возникновения пожара:

- периодическая проверка проводки;
- отключение оборудования при покидании рабочего места;
- проведение с работниками инструктажа по пожарной безопасности.

Для увеличения устойчивости помещения к ЧС необходимо устанавливать системы противопожарной сигнализации, реагирующие на дым и другие продукты горения. Оборудовать помещение огнетушителями, планами эвакуации, а также назначить ответственных за противопожарную безопасность. Согласно НПБ 166-97 [43] необходимо проводить своевременную проверку огнетушителей. Два раза в год (в летний и зимний период) проводить учебные тревоги для отработки действий при пожаре.

Одними из наиболее вероятных видов чрезвычайных ситуаций являются пожар, а также взрыв на рабочем месте.

Всякий работник при обнаружении пожара должен:

1. незамедлительно сообщить об этом в пожарную охрану;
2. принять меры по эвакуации людей, каких-либо материальных ценностей согласно плану эвакуации;
3. отключить электроэнергию, приступить к тушению пожара первичными средствами пожаротушения.

Учебные аудитории 10 корпуса ТПУ оснащены ручными углекислотными огнетушителями ОУ-2 по одному на аудиторию, а также аптечками первой помощи согласно требованиям ГОСТ Р 51057-01 [44].

Рабочее место располагается в 10 корпусе ТПУ в 117 аудитории. План эвакуации предоставлен на рисунке 35.

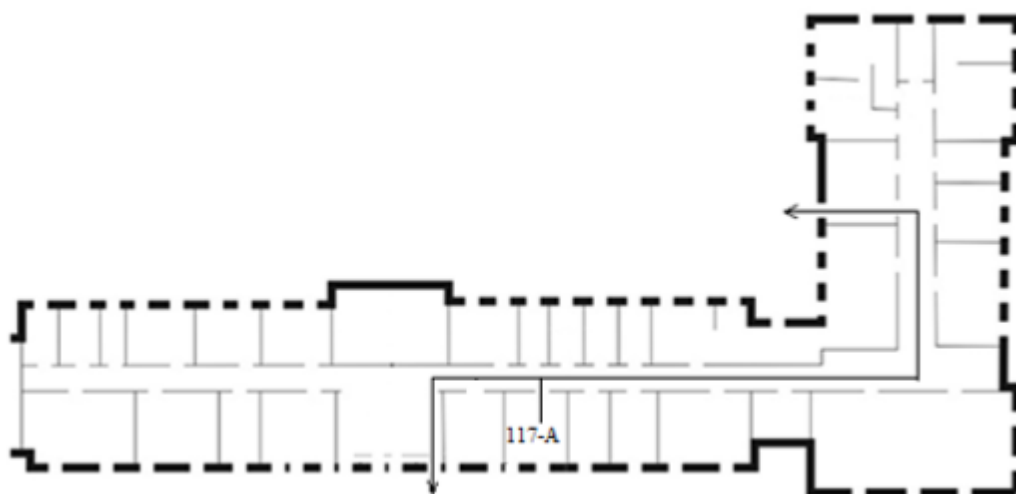


Рисунок 35 – План эвакуации при пожаре и других ЧС из помещений учебного корпуса №10, пр. Ленина, 2, 1-й этаж

Выводы по разделу

В результате проделанной работы можно сделать вывод, что социальная ответственность понимается как объективная необходимость нести ответственность за нарушение социальных норм. Он основан на социальной природе человеческого поведения.

Будущий специалист должен проводить профессиональную деятельность с учетом социальных, правовых, экологических и культурных аспектов, вопросов здоровья и безопасности, нести социальную ответственность за принимаемые решения, осознавать необходимость устойчивого развития.

В этом разделе, в соответствии с полученными знаниями, была продемонстрирована способность анализировать характер действия разработанных в работе решений. Социальная ответственность при разработке новых решений обеспечивает: устранение аварий; защиту здоровья работников; снижение вредного воздействия на окружающую среду; экономное использование не возобновляемых природных ресурсов.

Заключение

В ходе выпускной квалификационной работы были рассмотрены особенности разработки и тестирования программного обеспечения для сбора данных и диспетчерского контроля с применением WEB-технологий. Разрабатываемая WEB-SCADA доступна на рынке SCADA-систем и представляет собой интеграцию WEB-технологий и систем управления производством в промышленности.

В результате исследования протестирована разрабатываемая SCADA-система с помощью различных видов тестирования. Благодаря тестированию минимизируется риск возникновения ошибок в режиме работы продукта. Такой этап разработки ПО, как тестирование, снижает риски нарушения работы продукта, чем помогает в экономии ресурсов предприятия.

Завершающим этапом работы было проведение автоматизированного интегрального тестирования ПО WEB-SCADA с использованием инструмента Hermione для написания автоматизированных тестов на JavaScript. Такой инструмент позволяет значительно ускорить автоматизацию процесса запуска тестов, тем самым сократить время на тестировании.

Список публикаций и выступлений

1. Обложенко А.Д., Кочегурова Е.А. Сравнительный анализ и тенденции развития SCADA-систем для нефтегазовой промышленности // Молодежь и современные информационные технологии : сборник трудов XVII Международной научно-практической конференции студентов, аспирантов и молодых учёных, 17–20 февраля 2020 г., г. Томск / Национальный исследовательский Томский политехнический университет, Инженерная школа информационных технологий и робототехники ; под ред. Д. М. Сонькина [и др.]. — Томск: Изд-во ТПУ, 2020. – С. 318-319.
2. Обложенко А.Д., Кочегурова Е.А. WEB-технологии и тенденции развития SCADA-систем в сфере АСУ ТП // Современные технологии, экономика и образование: сборник материалов Всероссийской научно-методической конференции, 27–29 декабря 2029 г., г. Томск: Изд-во ТПУ, 2019. – С. 15-17.
3. Обложенко А.Д. WEB-технологии в сфере АСУ ТП // III Международная научно-техническая конференция молодежи ПАО «Транснефть», 19 ноября 2019 г., г. Томск

Список использованных источников

1. Введение в WEB-технологии // Webonto [Электронный ресурс]. - Режим доступа: <https://webonto.ru/vvedenie-v-web-tehnologii>.
2. Веб-технологии для разработчиков // MDN web docs [Электронный ресурс]. - Режим доступа: <https://developer.mozilla.org/ru/docs/Web>.
3. Языки разметки WEB-страниц // Bourabai [Электронный ресурс]. - Режим доступа: <http://bourabai.ru/dhtml/index.html>.
4. Основы сценариев HTML // Bourabai [Электронный ресурс]. - Режим доступа: <http://bourabai.ru/dhtml/index.html>.
5. Введение в JavaScript [Электронный ресурс]. - Режим доступа: <https://learn.javascript.ru/intro>.
6. Обзор JavaScript // MDN web docs [Электронный ресурс]. - Режим доступа: https://developer.mozilla.org/ru/docs/Web/JavaScript/JavaScript_technologies_overview.
7. Quasar Framework — универсальный Vue-фреймворк // Medium [Электронный ресурс]. - Режим доступа: <https://medium.com/devschacht/quasar-framework>.
8. Полезный инструмент для Vue-разработчиков // Хабр [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/company/ruvds/blog/459300/>.
9. Пользовательские сценарии // Лекция Яндекса [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/company/yandex/blog/346608/>.
10. Автоматическое тестирование с использованием фреймворка Mocha // Учебник JavaScript [Электронный ресурс]. - Режим доступа: <https://learn.javascript.ru/testing-mocha>.
11. Промышленный интернет вещей (IIOT) // Hewlett Packard [Электронный ресурс]. - Режим доступа: <https://www.hpe.com/ru/ru/what-is/industrial-iiot.html>.
12. Обложенко А.Д. WEB-технологии в сфере АСУ ТП // III Международная научно-техническая конференция молодежи ПАО

«Транснефть», 19 ноября 2019 г., г. Томск

13. Андреев Е.Б. SCADA-системы: взгляд изнутри / Е.Б. Андреев, Н.А. Куцевич, О.В. Синенко. – М.: РТСофт, 2004. – 176 с.
14. Supervisory Control And Data Acquisition [Электронный ресурс]. - Режим доступа: <http://www.tadviser.ru/index.php/SCADA>.
15. SCADA система MasterSCADA [Электронный ресурс]. - Режим доступа: <https://insat.ru/products/?category=9>.
16. SCADA Trace Mode [Электронный ресурс]. - Режим доступа: <http://www.adastra.ru/>.
17. Wonderware InTouch // Wonderware [Электронный ресурс]. - Режим доступа: <https://www.wonderware.ru/hmi-scada/intouch/>.
18. RSView // Компания Эско Восток КОМ [Электронный ресурс]. - Режим доступа: <https://www.eskovostok.ru/catalog/software/capacity/rsview>.
19. Обложенко А.Д., Кочегурова Е.А. WEB-технологии и тенденции развития SCADA-систем в сфере АСУ ТП // Современные технологии, экономика и образование: сборник материалов Всероссийской научно-методической конференции, 27–29 декабря 2029 г., г. Томск: Изд-во ТПУ, 2019. – С. 15-17.
20. Обложенко А.Д., Кочегурова Е.А. Сравнительный анализ и тенденции развития SCADA-систем для нефтегазовой промышленности // Молодежь и современные информационные технологии : сборник трудов XVII Международной научно-практической конференции студентов, аспирантов и молодых учёных, 17–20 февраля 2020 г., г. Томск / Национальный исследовательский Томский политехнический университет, Инженерная школа информационных технологий и робототехники ; под ред. Д. М. Сонькина [и др.]. — Томск: Изд-во ТПУ, 2020. – С. 318-319.
21. SmartPTT SCADA/IOT // SmartPTT [Электронный ресурс]. - Режим доступа: <https://smartptt.com/ru/solutions/scada/>.
22. Определение тестирования программного обеспечения // 4stud.info

[Электронный ресурс]. - Режим доступа: <http://www.4stud.info/software-construction-and-testing/lecture7.html>.

23. Тестирование. Фундаментальная теория // Хабр [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/post/279535/>.

24. Из чего состоит тестирование // Software-Testing [Электронный ресурс]. - Режим доступа: <https://software-testing.ru/library/testing/general-testing/2576-so-what-is-software-testing>.

25. Уровни тестирования // PSV Studio [Электронный ресурс]. - Режим доступа: <https://www.viva64.com/ru/t/0093/>.

26. Юзабилити-тестирование // QAlight [Электронный ресурс]. - Режим доступа: <https://qalight.com.ua/baza-znaniy/yuzabiliti/>.

27. Тестирование пользовательского интерфейса // QA evolution [Электронный ресурс]. - Режим доступа: <https://qaevolution.ru/testirovanie-po/vidy-testirovaniya-po/testirovanie-polzovatelskogo-interfejsa/>.

28. Нагрузочное тестирование // PerformanceLab [Электронный ресурс]. - Режим доступа: <https://www.performance-lab.ru/blog/load-testing/testirovanie-proizvoditelnosti>.

29. Тестирование безопасности или Security and Access Control Testing // Testing.ru [Электронный ресурс]. - Режим доступа: <http://www.protesting.ru/testing/types/security.html>.

30. Автоматизированное тестирование // QAlight [Электронный ресурс]. - Режим доступа: <https://qalight.com.ua/baza-znaniy/ruchnoe-i-avtomatizirovannoe/>.

31. Тестируем пользовательские сценарии вместе с «Гермионой» // Хабр [Электронный ресурс]. - Режим доступа: <https://habr.com/ru/company/yandex/blog/346608/>.

32. Панин В.Ф., Сечин А.И., Федосова В.Д. Экология для инженера://под ред. Проф. В.Ф. Панина. – М: Изд. Дом «Ноосфера», 2000. –284с.

33. ГОСТ 12.2.032-78. «Система стандартов безопасности труда. Рабочее место при выполнении работ сидя. Общие эргономические требования».

34. ГОСТ 12.2.061-81 ССБТ. «Оборудование производственное. Общие требования безопасности к рабочим местам».
35. СанПиН 2.2.2/2.4.1340-03. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».
36. ГОСТ 12.0.003-74. «Опасные и вредные производственные факторы. Классификация».
37. СанПиН 2.2.2/2.4.1340-03. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».
38. СанПиН 2.2.4.548–96. «Гигиенические требования к микроклимату производственных помещений».
39. СанПиН 2.2.2/2.4.1340-03. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».
40. СанПиН 2.2.4.1191–03. «Электромагнитные поля в производственных условиях»
41. ГОСТ Р 12.1.019-2009 ССБТ. «Электробезопасность. Общие требования и номенклатура видов защиты».
42. СНиП 2.01.02-85. «Противопожарные нормы».
43. НПБ 166-97. «Пожарная техника. Огнетушители. Требования к эксплуатации».
44. ГОСТ Р 51057-01. «Техника пожарная. Огнетушители переносные. Общие технические требования. Методы испытаний».

Приложение А
(справочное)

Раздел 2 SCADA-systems

Студент:

Группа	ФИО	Подпись	Дата
8BM81	Обложенко Алёна Дмитриевна		

Руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Кочегурова Е. А.	к.т.н., доцент		

Консультант-лингвист отделения иностранных языков ШБИП:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИЯ ШБИП	Сидоренко Т. В.	к.п.н.		

2 SCADA-systems

Today the problem of accurate and timely control of technological processes in many industrial enterprises and in the industry is relevant.

The world community has invented supervisory control and data collection systems (called SCADA systems) to implement a solution to this problem.

Currently, SCADA (Supervisory Control And Data Acquisition) is a mandatory attribute of an automated supervisory control system. The hardware platform for SCADA-systems are server and workstations based on industrial computers, panel computers and operator panels.

SCADA-systems solve the following tasks:

- a) data exchange through special drivers of communication devices in real time,
- b) algorithmic or logical control,
- c) information processing in real-time,
- d) display of the current technological process and other necessary information on the monitor screen in a convenient form for human perception,
- e) database maintenance in real time,
- f) alarm and control,
- g) preparation of reports on the progress of the process,
- h) network interaction between SCADA and industrial facilities,
- i) communication with external applications (DBMS, spreadsheets, etc.),
- j) analysis and forecasting of parameters coming from measuring instruments.

The first SCADA systems appeared in the USA in the 60s of the last century. However, the most significant development of the system occurred in the 70-80s of the XX century due to the development of hardware element base (microprocessor technology). The main functionality of SCADA systems of that time corresponded to the capabilities of the first control computers. Modern SCADA systems are well structured and ready to use.

The development of SCADA systems is a new approach to the problems of the

human factor in control systems, aimed at a person (operator / dispatcher), his tasks and the functions performed by him.

2.1 Existing SCADA-systems

2.1.1 SCADA «MasterSCADA» of ИИCAT company (Russia)

Currently, the world market offers many product solutions implemented in the form of SCADA-systems. Next, the most popular SCADA systems will be considered.

SCADA-system "MasterScada" - a system for process control systems, solving accounting and dispatching problems, the leader of the Russian market.

The main features of the MasterScada software package are scalability, modularity and an object-oriented development approach. The package implements tools and methods for developing projects that provide a sharp reduction in labor costs and increase the reliability of the created system. This is the first system in Russia where an object approach to development is implemented.

MasterScada is a full-featured SCADA software package with extensible functionality. The package is built on a client-server architecture with the ability to work on local networks and on the Internet. In addition to working with the upper level, the system allows to program controllers with an open architecture. The reception and transmission of data and messages based on OPC standards are built into the core of the package. Open descriptions of interfaces and data formats, as well as maximum support for all standards (XML, HTML, ODBC, OLE, COM / DCOM, ActiveX, etc.) provide all the necessary capabilities for interacting with external programs and systems. Figure 7 shows a screenshot of the development of the mnemonic diagram in MasterSCADA.

The following are the key benefits of MasterScada for creating process control systems:

- a) a unified development environment for automated control systems,
- b) separate configuration of the logical structure of the object,
- c) openness and compliance with standards,
- d) intuitive ease of development,

- e) convenience of tools,
- f) convenience of development methods,
- g) unlimited flexibility of computing capabilities,
- h) powerful three-dimensional graphics and multimedia,
- i) object approach.

Companies that use this system: Rosneft, Gazprom, Lukoil and others. The total number of Master SCADA system implementations is tens of thousands in almost all industries in Russia, near and far abroad.

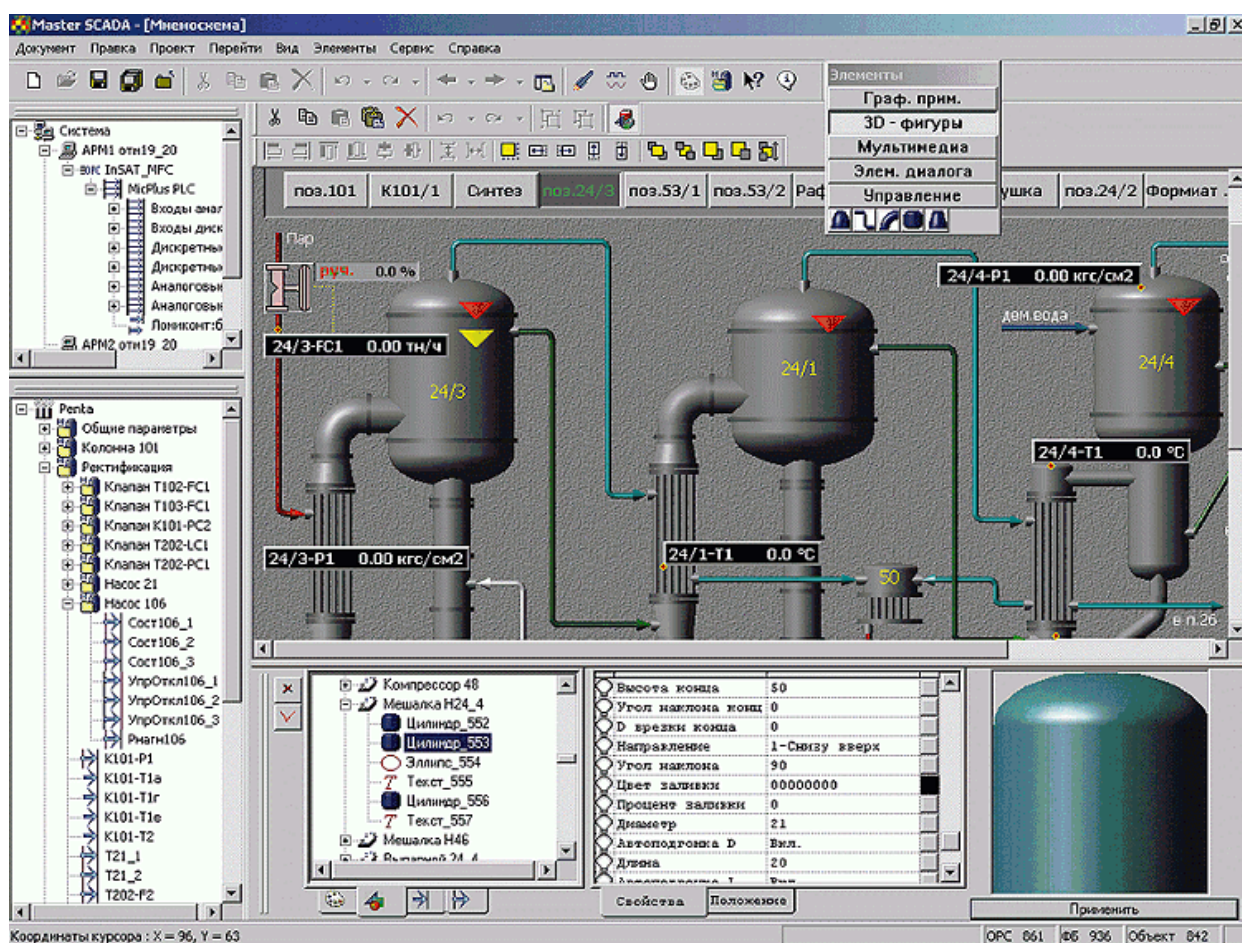


Figure 7 – Example of the development of a mnemonic circuit in SCADA
MasterSCADA

2.1.2 SCADA «Trace Mode» of AdAstrA Research Group (Russia)

SCADA "Trace Mode" - a software package of the SCADA HMI class, which is designed to develop programs for industrial control systems, telemechanics systems. Currently, the most purchased domestic software system in Russia for the automation of technological processes.

The system is focused on working with controllers and supports all modern information technologies. To connect controllers and third-party systems, you can use OPC technology (Open Platform Communications). ODBC (Open Database Connectivity) is used to access databases.

Advantages of the Trace Mode system:

- a) all built-in drivers (for more than 2586 PLCs) are delivered immediately and free of charge,
- b) a single controller programming tool,
- c) electronic documentation and integrated help system,
- d) scalability from 16 to 1,000,000 input / output points,
- e) development tools for telemechanics systems,
- f) the fastest real-time system,
- g) a free set of drivers for input / output devices,
- h) high reliability of the system (AWP with shock-free restart),
- i) TRACE MODE demonstration projects (examples of automation systems based on process simulation),
- j) a large library of finished components (more than 1000 pcs.),
- k) adaptive self-tuning of PID controllers, which is based on Russian patented technology.

The dynamic characteristics and reliability of the software allow the use of developed automation systems in industries such as petrochemistry, metallurgy, mechanical engineering, as well as in research.

Trace Mode 6 software products are divided into an integrated development environment and executive modules. The first is installed at the workplace of a development engineer and is intended to create an automation system. Executive

modules are designed to work on an existing automation facility.

A feature of this software product is also the “technology of a single programming line” - this is the ability to develop all ACS modules using one tool. In one project, technology allows you to create a human-machine interface, a resource accounting system, program industrial controllers and develop a web interface.

Figure 8 shows a screenshot of the mnemonic design in Trace Mode 6.

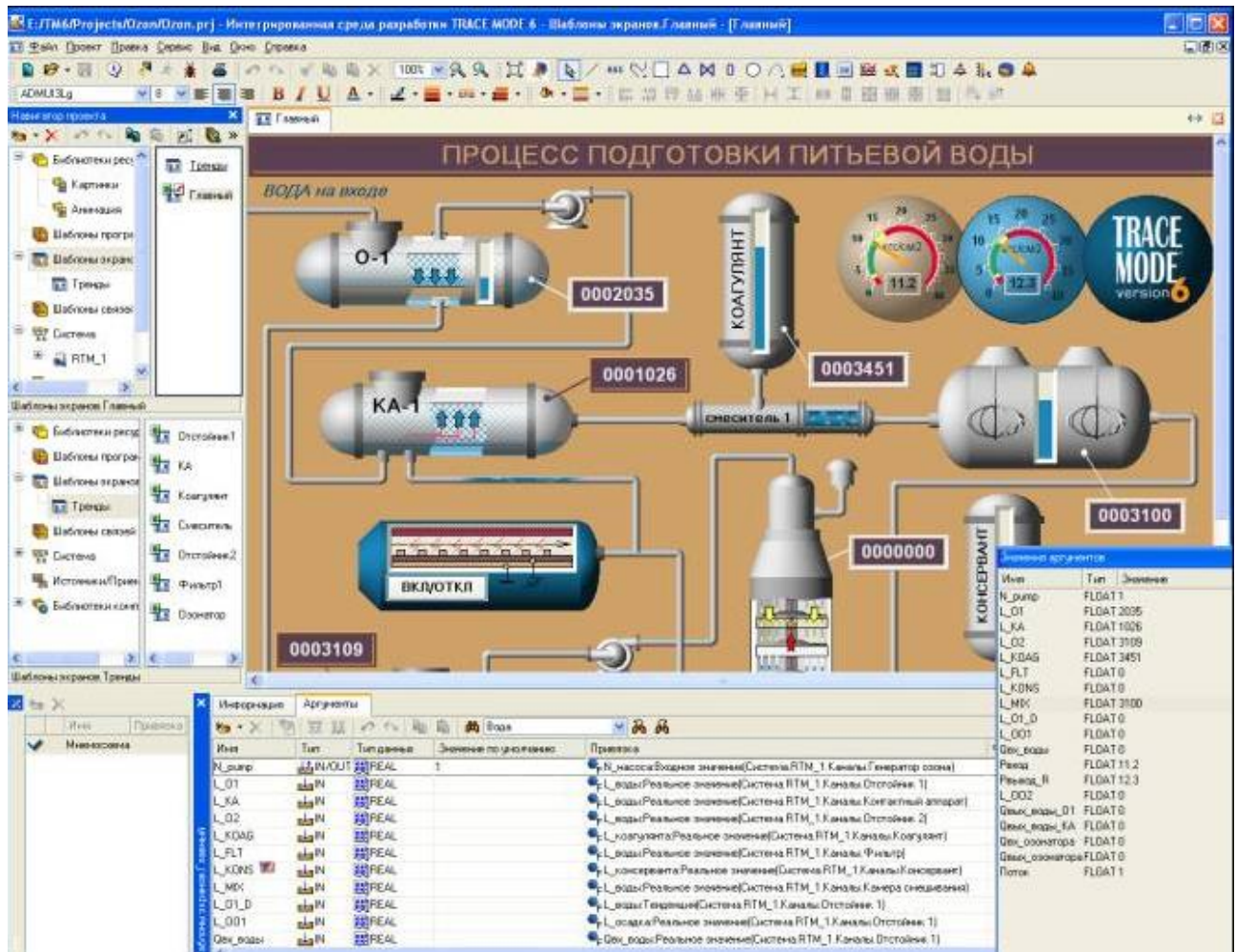


Figure 8 – Example of the development of a mnemonic circuit in SCADA
«Trace Mode»

2.1.3 SCADA «InTouch» of Wonderware company (USA)

Wonderware InTouch HMI is software for industrial automation, visualization and process control. This product is one of the leaders in the Russian and foreign markets of SCADA products.

The InTouch software package contains two components - a development environment and a runtime environment. In the development environment, mnemonic schemes are created, input and output signals and parameters are determined, control algorithms are developed and operator rights are assigned. Such an application operates in a runtime environment.

In order for the application to exchange data with the equipment, you must use a separate program. Typically, an input / output server is intended for use with a specific type of equipment, such as industrial controllers.

The main tasks solved using InTouch:

- a) collection of signals (determination of the state of the production process at the current point in time - temperature, pressure, position, etc.),
- b) automatic monitoring of the state of monitored parameters, the generation of alarms and the issuance of messages to the operator in graphical and text form,
- c) tracking operator's actions, registering it in the system using a username and password, assigning certain access rights,
- d) output of control actions on industrial controllers and actuators to regulate continuous or discrete processes,
- e) automatic registration of events, in which changes in production parameters are recorded, it is possible to view recorded data in graphical form,
- f) product quality control by statistical processing of recorded parameters,
- g) report generation.

Figure 9 shows a screenshot of the development of the mnemonic diagram in InTouch.

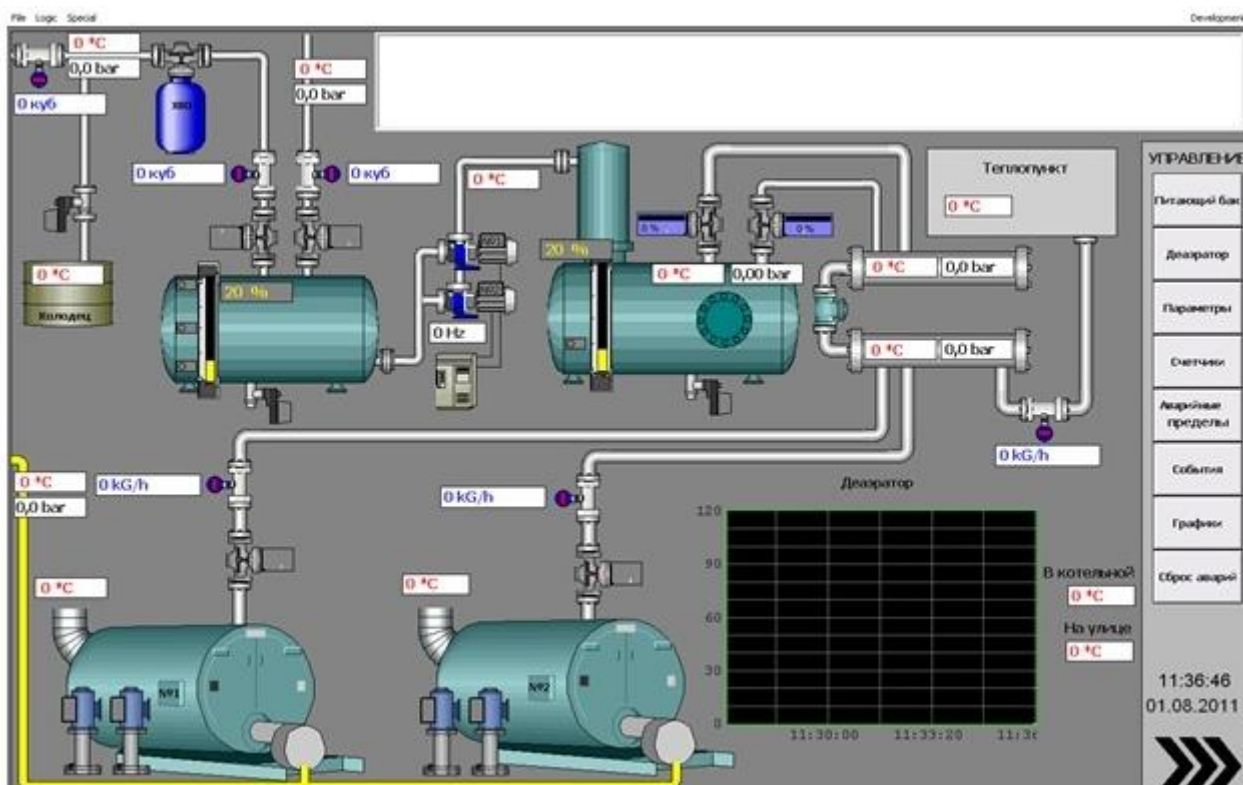


Figure 9 – Example of the development of a mnemonic circuit in SCADA InTouch

2.1.4 SCADA «RSView32» of Rockwell Automation company (USA)

RSView 32 is an integrated software that is based on components of a human-machine interface for monitoring and process control. The presented SCADA-system expands the possibilities of process overview thanks to open communication standards that ensure uninterrupted connection to other Rockwell Software products, Microsoft products, as well as third-party applications.

RSView32 is the first human-machine interface software product that fully utilizes Microsoft's advanced technology.

This advantage is intended to enable:

- open graphic displays in the form of OLE containers for ActiveX controls,
- develop an object model to reveal part of its basic functions for RSView32 interaction with other software products,
- Integrate Microsoft Visual Basic for Applications (VBA) for unlimited project customization,

d) maintain OPC standards for server and client for security and fast communication with a wide range of equipment,

e) Implementation of Add-On Architecture (AOA) technology to expand RSView32 functionality and integrate new functions directly into the RSView32 core.

The advantages of using this SCADA system are as follows:

- a) high reliability of hardware and software,
- b) intuitive user interface,
- c) use only open communication standards,
- d) compatibility with Allen Bradley hardware platforms,
- e) An open and flexible DNA-Manufacturing Microsoft-based architecture for manufacturing.

Figure 10 shows a screenshot of the development of the mnemonic diagram in RSView32.

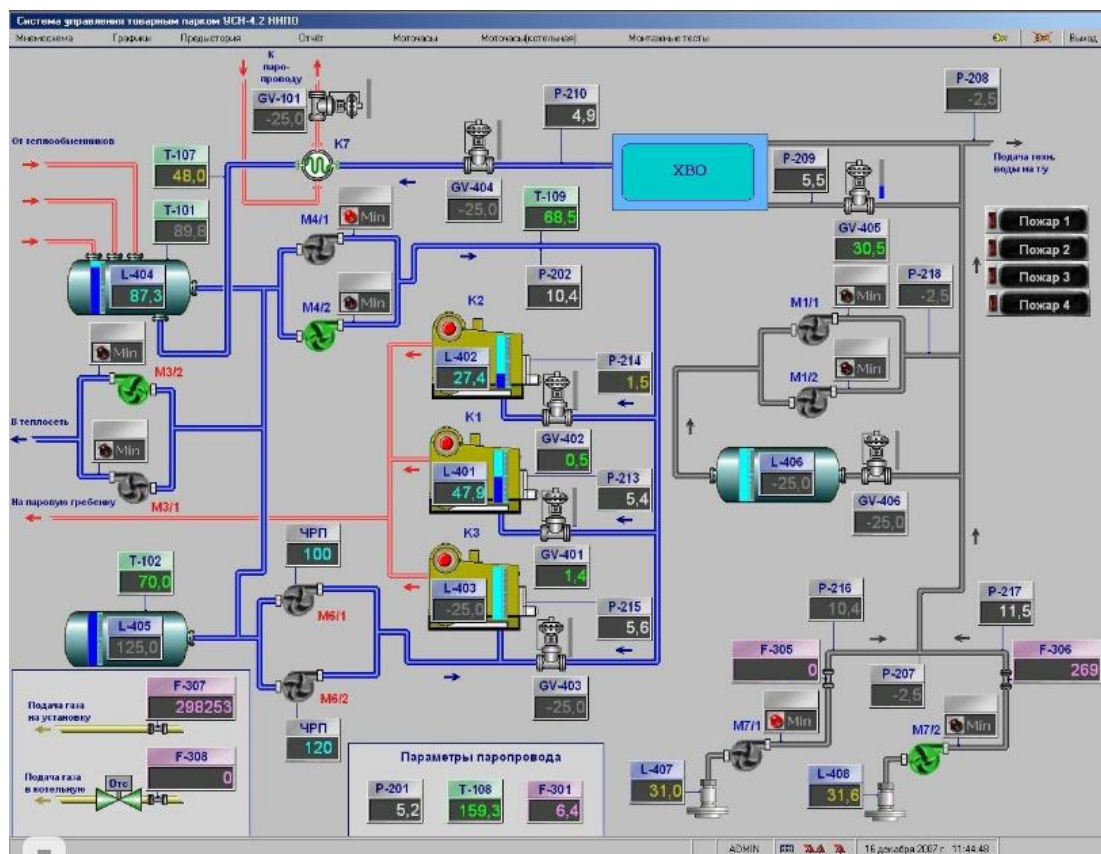


Figure 10 – Example of the development of a mnemonic circuit in SCADA RSView32

2.2 Proposed solution

Most of the presented SCADA-systems need significant changes to move into the world of WEB. However, support for such products is complicated by the interaction of software components with each other.

In this master's thesis, the proposed SCADA-system is the introduction of WEB-technologies into the automation system, which means the possibility of remote access to data and controls in real time at anytime from anywhere. Some elements of such systems existed earlier, but only with the advent of WEB-technologies did a real and inexpensive solution appear that made it possible to "make visible" the ACS TP system and remotely control it. The implementation of such a system implies the presence of a powerful system of information security and delimitation of user access rights, which is already present in many modern SCADA software packages.

The advantage of using Internet technologies in industrial control systems is the ability to use any web browser on the computer's dispatcher, regardless of manufacturer, type of hardware platform or operating system.

Management and monitoring via the Internet is carried out from anywhere in the world using a computer or mobile phone. Such an opportunity is especially important for senior management.

Other advantages of automated systems using the Internet are:

- a) reduction in the cost of functioning of industrial control systems due to remote control,
- b) lower maintenance costs due to remote diagnostics, debugging and software updates via the Internet,
- c) the ability to monitor the status of the production or technological process,
- d) the ability to automatically call the emergency service in case of triggering of gas sensors, smoke, flame, flooding, etc.,
- e) A wide selection of ready-made technical solutions, hardware and software products for working with the Internet.

Conclusion

During the final qualification work, the features of the development and testing of software for data collection and supervisory control using WEB technologies were considered. The developed WEB-SCADA is available on the SCADA-systems market and represents the integration of WEB-technologies and industrial production control systems.

As a result of the study, the developed SCADA system was tested using various types of testing. Thanks to testing, the risk of errors in the operating mode of the product is minimized. This stage of software development, such as testing, reduces the risks of disruption of the product, which helps in saving enterprise resources.

The final stage of work was the conduct of automated integrated testing of WEB-SCADA software using the Hermione tool for writing automated JavaScript tests. Such a tool can significantly accelerate the automation of the test launch process, thereby reducing testing time.